SCHOOL OF
CIVIL ENGINEERING

INDIANA
DEPARTMENT OF HIGHWAYS

JOINT HIGHWAY RESEARCH PROJECT

FHWA/IN/JHRP-84/11

Interim Report

DEVELOPMENT AND USE OF A MANAGEMENT INFORMATION SYSTEM TO IDENTIFY AREAS OF ROUTINE MAINTENANCE PRODUCTIVITY IMPROVEMENT

V. Alan Sanderson
Kumares C. Sinha

PURDUE UNIVERSITY

SCHOOL OF CIVIL ENGINEERING

INDIANA DEPARTMENT OF HIGHWAYS

JOINT HIGHWAY RESEARCH PROJECT

FHWA/IN/JHRP-84/11

Interim Report

DEVELOPMENT AND USE OF A MANAGEMENT INFORMATION SYSTEM TO IDENTIFY AREAS OF ROUTINE MAINTENANCE PRODUCTIVITY IMPROVEMENT

V. Alan Sanderson
Kumares C. Sinha

PURDUE UNIVERSITY

[redacted]

JOINT HIGHWAY RESEARCH PROJECT

FHWA/IN/JHRP-84/11

Interim Report

DEVELOPMENT AND USE OF A MANAGEMENT INFORMATION SYSTEM TO IDENTIFY AREAS OF ROUTINE MAINTENANCE PRODUCTIVITY IMPROVEMENT

V. Alan Sanderson
Kumares C. Sinha

Interim Report

DEVELOPMENT AND USE OF A MANAGEMENT INFORMATION SYSTEM TO IDENTIFY AREAS OF ROUTINE MAINTENANCE PRODUCTIVITY IMPROVEMENT

To: H.L. Michael, Director
Joint Highway Research Project

July 3, 1984
Project: C-36-67N
File: 9-11-14

From: K.C. Sinha, Research Engineer
Joint Highway Research Project

Attached is the Interim Report on the HPR Part II Study entitled, "Development and Use of a Management Information System to Identify Areas of Routine Maintenance Productivity Improvement." The research has been conducted by Mr. V. Alan Sanderson, Graduate Research Assistant under the direction of Professor K.C. Sinha, Research Engineer.

The report here is primarily concerned with Tasks 1 through 5 of the work plan and it presents the development of a monitoring procedure for evaluating and improving productivity of maintenance operations. Field observations are currently underway and the Final Report will document the remaining tasks.

This report is forwarded to the IDOH & FHWA as nartial fulfillment of the objectives of the research.

Respectfully submitted,

KCSinha

K.C. Sinha
Research Engineer

KCS/bjm

cc: A.G. Altschaeffl
J.M. Bell
W.F. Chen
W.L. Dolch
R.L. Eskew
J.D. Fricker
J.R. Skinner
W.H. Goetz
G.K. Hallock
J.F. McLaughlin
R.D. Miles
P.L. Owens
B.K. Partridge
G.T. Satterly
C.F. Scholer
R.M. Shanteau
K.C. Sinha
C.A. Venable
L.E. Wood
S.R. Yoder

Interim Report

DEVELOPMENT AND USE OF A MANAGEMENT INFORMATION SYSTEM TO IDENTIFY AREAS OF ROUTINE MAINTENANCE PRODUCTIVITY IMPROVEMENT

by

V. Alan Sanderson
Graduate Research Assistant

and

Kumares C. Sinha
Professor of Civil Engineering

Joint Highway Research Project

Project No.: C-36-67N
File No.: 9-11-14

Prepared as Part of an Investigation

Conducted by

Joint Highway Research Project
Engineering Experiment Station

Purdue University

in cooperation with the

Indiana Department of Highways

and the

U.S. Department of Transportation
Federal Highway Administration

The contents of this report reflect the views of the authors who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views or policies of the Federal Highway Administration. The report does not constitute a standard, specification, or regulation.

Purdue University
West Lafayette, Indiana

July 3, 1984

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No. FHWA/IN/JHRP-84-11	2. Government Accession No.	3. Recipient's Catalog No.

4. Title and Subtitle DEVELOPMENT AND USE OF A MANAGEMENT INFORMATION SYSTEM TO IDENTIFY AREAS OF ROUTINE MAINTENANCE PRODUCTIVITY IMPROVEMENT .	5. Report Date July 3, 1984 6. Performing Organization Code
7. Author(s) V. Alan Sanderson and Kumares C. Sinha	8. Performing Organization Report No. JHRP-84-11
9. Performing Organization Name and Address Joint Highway Research Project Civil Engineering Building Purdue University West Lafayette, Indiana 47907	10. Work Unit No. 11. Contract or Grant No. HPR-1(21) Part II
12. Sponsoring Agency Name and Address Indiana Department of Highways State Office Building 100 North Senate Avenue Indianapolis, Indiana 46204	13. Type of Report and Period Covered Interim Report Tasks 1 through 5 14. Sponsoring Agency Code

15. Supplementary Notes Prepared in cooperation with the U.S. Department of Transportation, Federal Highway Administration on HPR Part II Study "Development and Use of A Management Information System to Identify Areas of Routine Maintenance Productivity Improvement".

16. Abstract

This report presents the development of a monitoring procedure that can provide the maintenance managers of the Indiana Department of Highways with a relatively simple and straight-forward base for evaluating and improving productivity of maintenance operations. The procedure involves calculation of average cost of performing a given maintenance activity to identify "deviate" subdistricts (those with very high or low costs), analysis of labor and material use information, review of equipment records, and field observations of crews at work in order to determine the reasons for the deviate cost.

17. Key Words Routine Maintenance; Management Information System; Productivity	18. Distribution Statement No restrictions. This document is available to the public through the National Technical Information Service, Springfield, Virginia 22161

19. Security Classif. (of this report) Unclassified	20. Security Classif. (of this page) Unclassified	21. No. of Pages 278	22. Price

ACKNOWLEDGMENTS

Many individuals assisted in this project. In particular, the assistance of Mr. R. Clay Whitmire of the Indiana Department of Highways and Mr. Donald Johnson of the Federal Highway Administration is gratefully acknowledged.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

APPENDICES

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (Continued)

Appendix
Figure

EXECUTIVE SUMMARY

During the last several years, funding highway maintenance operations has become increasingly difficult. The outlay required to maintain the highway system has increased, but the money available for routine maintenance has not kept pace with the need.

An effective management approach is essential in planning and controlling highway activites, particularly in an era of limited resources. The wide use of computers has greatly contributed to the development of maintenance management systems. A large amount of information can now be processed to distill the necessary information for the highway managers. However, most of the available procedures for pursuing maintenance management reports are generally too detailed and complex for easy and ready use by the managers.

In this study, a monitoring procedure was developed to provide Indiana Department of Highways maintenance managers with a relatively simple and straight-forward base for evaluating and improving productivity of maintenance operations. The use of this procedure can assist maintenance managers in improving the productivity of their operations. Using information currently collected by IDOH, a computer program prepared as part of the study identifies subdistricts with unusually high or low average cost. Further study of these subdistricts should reveal the reasons for their deviations in resource use. The need for corrective measures can be identified, or unusually efficient methods may be identified and spread throughout the state, resulting in improved productivity.

The procedure involves calculation of average cost of performing a given maintenance activity to identify "deviate" subdistricts (those with very high or low costs), analysis of labor and material use information, review of equipment records, and field observations of crews at work in order to determine the reasons for the deviate cost. Its use was demonstrated with two activities, crack sealing and shallow patching.

CHAPTER 1

INTRODUCTION

During the last several years, funding highway maintenance operations has become increasingly difficult. The outlay required to maintain the highway system has increased due to inflation plus the fact that the facilities are aging. Furthermore, although the amount of Federal matching funds has increased as a result of the 1982 Surface Transportation Assistance Act, no Federal money is available for routine maintenance. In order to make use of the available Federal funds, a state has to provide an increased amount of state money, thus causing a serious squeeze on state funds that are necessary for routine maintenance. These factors have made state highway officials more concerned than ever about accountability and the effective use of the resources at their disposal. Heightened citizen awareness of highway costs and services, and rising costs along with tighter constraints on allocation, have required cutbacks and changes in many highway programs. In general, the shifting of the fiscal environment surrounding state highway programs has made programming and budgeting a more critical process. In view of the limited funds available, it is clear that increased effort must be directed toward using maintenance resources as efficiently as possible. To accomplish this, highway managers must be able to identify areas where resources are not being used efficiently as well as to determine corrective measures.

The need to concentrate efforts on the highway maintenance program is due to a number of reasons. First, maintenance is a large program in terms of both manpower and expenditures. Second, the difficulty in recent years, of financing the maintenance program at a level consistent with needs, has created a growing backlog of estimated maintenance deficiencies. Third, the maintenance program has high public visibility. Fourth, an effective maintenance management program can greatly improve the efficiency of the program.

An effective management approach is essential in planning and controlling highway activities. Efficiency measures are useful for planning and control on most managerial levels of an organization. Maintenance management, pavement management, and project management systems are being increasingly used by highway managers. Much information exists in the area of pavement management systems. Several states have already implemented or are in the process of implementing a pavement management system. Indiana is one of the states that are currently involved in the development of a management system of its own. A pavement management system is intended to provide systematic and objective information regarding the optimal pavement maintenance strategy on a project-by-project basis. These projects are generally of major maintenance type involving capital investment. A routine maintenance management system, on the other hand, involves all forms of highway routine maintenance, including pavement [1,4,9,11]. A routine maintenance management system is directed at the overall maintenance program, and it involves the preparation of quality and quantity standards, road inventory, maintenance cost model, performance, budget, and feedback reports. The purpose of a maintenance management system is to contribute to the management functions of identifying total work backlog, ranking it, budgeting and updating the total system [9].

The experience gained in both pavement management and routine maintenance management systems indicates that efficient maintenance operations depend to a great degree on management factors, particularly the ability of the first- and second-line supervisors who are responsible for the planning and the application of the proper mix of personnel, equipment, materials and procedures to routine task and emergency requirements in the maintenance program [4].

The wide use of computers has greatly contributed to the development of maintenance management systems. A large amount of information can now be processed to distill the necessary information for the highway managers. However, many of the available procedures for pursuing maintenance management reports are generally too detailed and complex for easy and ready use of the managers. This is particularly true for the Indiana Department of Highways (IDOH). A need exists to identify the essential level of information required by routine maintenance managers and to develop procedures that can produce the necessary information in forms useful for improved control of routine maintenance activities.

Purpose of the Study

The IDOH has an excellent system of maintenance data collection in the form of crew day cards. However, the reports now available from the computer compilation of these data are much too detailed for highway managers to compare readily the productivity of various maintenance units and identify those with deviations, high or low, in resource use. The purpose of this study is to develop a monitoring procedure that can provide the necessary management information from the collected data in a systematic manner to provide a base for evaluating and improving productivity of maintenance operations.

With this procedure, maintenance personnel may be alerted to areas where resources are not being used efficiently or to areas with unusually high efficiency in resource use. Investigation of these areas may reveal a need for corrective action, an efficient innovative procedure, or important factors not reflected in crew day card data. This can lead to improved productivity by correcting problems as well as disseminating innovative techniques developed by a unit.

Organization of the Report

The report consists of five chapters and two appendices. Chapter 2 is a discussion of maintenance management systems and programming in general, and within the Indiana Department of Highways in particular. Chapter 3 is an analysis of the distribution of IDOH routine maintenance expenditures for the fiscal year 1981-82. The distribution by maintenance activity, by IDOH subdistrict, and the relations between expenditure and factors such as lane-miles are examined. Chapter 4 presents the procedure to be used to compare and evaluate the performance of routine maintenance by the subdistricts. The computer program developed in the study is explained, and examples of the use of the procedure are presented. Chapter 5 contains the summary and recommendations of the study. Appendix A is the user's manual for the computer program, and Appendix B is the listing of the computer program.

CHAPTER 2

MAINTENANCE MANAGEMENT INFORMATION SYSTEMS AND PROGRAMMING

The overall purpose of a maintenance management information system is to aid a highway agency in achieving its maintenance program goals and objectives.

Typical goals of a highway maintenance program include the following [11]:

1. Preserving the investment in roadways, bridges, and other highway components.

2. Achieving the necessary expenditure of resources in an efficient manner.

3. Optimizing the use of resources by avoiding an undue amount of idle labor and equipment.

A specific objective may be to reduce a routine maintenance backlog by a certain percentage, or to increase the amount of preventative maintenance by a given amount.

A maintenance management information system will help to achieve these goals and objectives by establishing a formal, rational method for estimating maintenance needs, determining resources required to meet these needs, providing a means to schedule efficiently allocation of labor, equipment, and materials throughout the year, and by providing feedback reports to monitor the production and productivity of maintenance forces in carrying out the maintenance program.

An ideal maintenance management information system may be divided into seven elements [2,3,9,10]:

1. The overall maintenance task should be divided into specific, defined activities. Examples of individual activities are patching, crack sealing, mowing, bridge repair, litter pickup, and so on.

2. A maintenance features inventory which lists the physical features of the highway requiring maintenance. Examples include the number of lane-miles of bituminous pavement, miles of unpaved ditch, linear feet of guardrail, and number of signs.

3. A maintenance needs survey which is used to estimate the level of maintenance needs.

4. A performance standard for each defined maintenance activity, which includes criteria for determining when that activity is required, the acceptable quality of work, the standard work method, expected production rate, and required resources.

5. Where maintenance needs are greater than available resources, a cost model may be used to determine a priority ranking of the various competing needs.

6. A scheduling and budgeting mechanism allows the preparation of an annual work program based on efficient allocation of resources, both physically throughout the state, and temporally throughout the year.

7. Feedback reports allow management to monitor the progress of maintenance forces in achieving the work program, to compare the performance of various maintenance districts, and to check the accuracy of their quantity standards. Appropriate reports could indicate such things as the use of improper methods, a need for training, or the improper scheduling of activities and allocation of resources.

Productivity Considerations

In order to manage effectively and efficiently a highway routine maintenance program, managers must be able to monitor the productivity or the performance of maintenance forces. By monitoring and comparing the productivity of various maintenance districts, maintenance managers will be able to identify areas where resources are not being used efficiently, and will be guided to investigate further and take corrective measures. On the other hand, highly efficient work methods may also be identified and transferred to other areas of the state.

Productivity can be represented by two broad measures: efficiency and effectiveness. Efficiency refers to how much work is produced with the given resources of labor, materials, and equipment. Effectiveness refers to how well the work accomplishes its objectives. In the case of the maintenance program, effectiveness refers to how well the program achieves its overall goals such as preserving the investment in the physical highway system and providing safe and efficient travel to system users. For a specific maintenance activity, effectiveness may refer to the quality of work performed.

To provide a meaningful measure of routine maintenance productivity, both efficiency and effectiveness must be considered. An efficient maintenance operation is not desirable if it is not effective, and an effective operation should be as efficient as possible.

Efficiency Measures

Efficiency may be measured in terms of a resource used per production unit or cost per production unit for a given maintenance activity. The number of manhours per production unit is one such measure. This would be suitable for maintenance activities which are measured in units other than manhours, such as shallow patching, where efficiency could be measured in manhours per ton of bituminous mixture placed. However, the production unit of many activities may be manhours, such as the case with brush cutting and sign maintenance. For these activities, an alternative efficiency measure would be necessary.

An alternative to manhours per production unit would be the amount of a given material per production unit. For example, the amount of herbicides used per manhour of herbicide treatment. However, a problem may arise in deciding which particular material to include in the efficiency measure when an activity involves the use of several materials. Furthermore, with many activities, the same mix of materials is not always used. If the material chosen as an efficiency measure is not always used, the usefulness of the measure is diminished.

A preferred approach to measuring efficiency is to calculate the average cost per production unit for an activity. This approach weights each component involved in performing maintenance, labor, materials, and equipment, with its unit cost. Poister, et al. state that cost per production unit is the most direct measure of efficiency, disregarding the quality of work [12]. However, the quality of the maintenance work should also be considered. In a study of routine maintenance in the Pennsylvania Department of Transportation, an inverse relation between efficiency and quality of work was found [12].

Quality Measures

In 1970, the Ohio Department of Transportation developed a system to measure the quality of routine highway maintenance. The three goals of the system were "to measure objectively the quality of maintenance achieved by highway maintenance forces, to establish acceptable standards of maintenance quality, and to provide a means of setting annual district and statewide maintenance performance objectives" [11].

The system involves inspections of randomly selected sections of highways throughout the state. A two-man crew inspects each section and counts the number of identifiable items requiring maintenance work. These data are converted to average numbers of work items per mile for each work category in each county. The numbers of "recordable conditions" per mile are plotted on bar charts along with the maintenance expenditure per lane-mile for the various work categories and counties. The charts are used by management at several levels for control of the maintenance program.

Fifteen recordable conditions which affect the physical integrity, safety, rideability, or aesthetics of the highway were defined objectively. For example, a shoulder drop-off of two inches or more which continues for more than six feet but less than 100 feet is one unit of the recordable condition "shoulder drop-off" [11].

Plots of recordable conditions per mile and expenditure per lane-mile have indicated that counties with higher expenditures tend to have fewer reportable conditions. These plots can be used by a district maintenance manager to monitor the performance of the county maintenance forces in his district, to monitor the change in maintenance effectiveness over time, to help a county supervisor plan work better, to acquaint him with a better method another county may be using, or to recognize the supervisor whose county has the fewest number of reportable conditions, thereby creating a sense of competition among the counties [11].

The Pennsylvania Department of Transportation has developed a Trained Observer System (TOS) patterned after the Ohio system to survey objectively the condition of highways with respect to physical and service features [12]. As in Ohio, randomly-selected highway segments are inspected, and the numbers of objectively-defined "reportable conditions" in various roadway elements are recorded. The TOS is intended to meet two goals. First, to improve the basis for allocation of maintenance funds by indicating where the greatest needs are, and second, to provide a mechanism for evaluating the overall effectiveness of the maintenance program [12].

A follow-up study was conducted in Pennsylvania to determine the relation between highway conditions measured by the TOS and the maintenance needs as determined by county maintenance managers. The study was to determine, the extent to which "production units or the dollar cost of maintenance work estimated for individual road sections [can] be explained by the counts of TOS reportable conditions on those sections" [14]. The study took survey sections from a recent TOS survey and had assistant county maintenance managers inspect the same sections and record the required maintenance. Estimated costs for the recommended maintenance work were calculated and correlated with TOS-reported conditions.

A surprising lack of correlation was found. The study discovered that rarely were specific maintenance actions consistently recommended in response to specific reportable conditions. Rather, "maintenance applications seem[ed] to be sporadic, perhaps more dependent upon the total mix of conditions on a given roadway and the resources available within each county" [14].

Several explanations were offered to explain this lack of correlation. There were reliability problems within the TOS with TOS teams missing a significant number of reportable conditions. Some definitions of TOS reportable conditions proved inadequate or inappropriate with respect to maintenance, with TOS observers counting conditions to which routine maintenance forces would not respond, such as rutting in wheel paths [14]. TOS observers would count each occurrence of a reportable condition within its 1/4-mile section, while the recommendation of maintenance managers would depend upon the concentration of the defect, the condition of the roadway on each side of the section, and the volume and class of traffic using the road.

There is a basic difference in the TOS survey and assessment by maintenance managers. TOS teams just record the defects seen in the highway, while maintenance managers also consider prevalence of the defect, the probable cause, the overall condition of the road, and the resources the county has. Thus, the recommended maintenance would not necessarily be expected to correspond closely with the "need" as evidenced by the TOS counts [14]. However, the gathering of data for this study was not strictly controlled, and it is not possible to conclude that there is no relation between TOS conditions and maintenance needs [14].

The quality of maintenance must be considered in an evaluation of routine maintenance productivity. A low unit cost for maintenance is not desirable if the reason for the low cost is a low level of quality in the work.

The Indiana Department of Highways does not have a formal comprehensive survey of the condition of its highways. Without a measure of this key element, it is difficult to monitor and to evaluate properly productivity of maintenance forces. Implementation of a survey such as those in Ohio and Pennsylvania could provide this key missing information, but in view of current staff and budget limitations, such a survey cannot be implemented.

An alternative for a comprehensive quality survey is to make spot checks of work quality in subdistricts that have been singled out for individual study. An efficiency analysis on the basis of cost can be used to identify subdistricts with unusually high and low costs, and then these subdistricts can be singled out for specific study of work quality.

Maintenance Programming In The IDOH

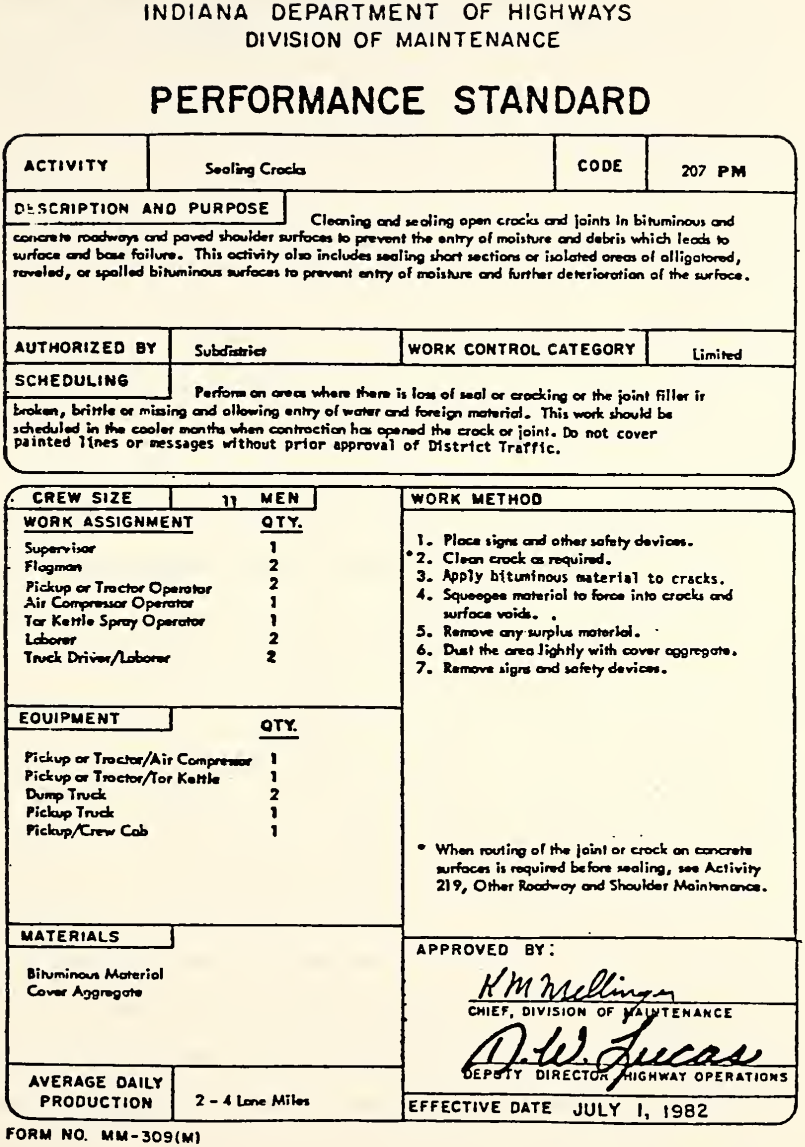
The Indiana Department of Highways Division of Maintenance has three levels of management involved in programming routine maintenance: the Central Office, the District, and the Subdistrict. The IDOH is divided into six districts, with each district divided into six, or in one case seven, subdistricts. Each subdistrict foreman is responsible for three or four units to which crews and equipment are assigned. It is the unit personnel who actually perform the physical work on the highway system.

The IDOH has developed a maintenance management information system for the programming, scheduling, and monitoring of routine maintenance operations. Individual activities have been defined and performance standards prepared. Figure 2.1 is an example of a performance standard.

A maintenance feature inventory was conducted for the entire highway system. For each routine maintenance activity, there are physical elements of the roadway on which that activity is performed. For example, the physical features that require shallow patching are lane-miles of low bituminous pavement, high bituminous pavement, bituminous over concrete pavement, and paved shoulder [7,8]. A low bituminous pavement refers to a highway that evolved through the years, such as a local path used by farmers that was gradually upgraded to a paved surface, and became a state highway. A high bituminous pavement refers to a highway that was formally planned, designed, and constructed as a state highway. The physical feature associated with joint and bump burning is the number of bituminous over concrete lane-miles, and the feature associated with cleaning and reshaping ditches is the number of miles of unpaved ditch [7,8].

The programming process begins with the subdistrict foremen who estimate their needs for each maintenance activity in the next fiscal year. This is based on a combination of foremen's visual inspection during the course of their daily work, and records of work performed in the previous year.

Several months before the start of the fiscal year, central office and district personnel visit with each subdistrict's personnel to discuss their estimated needs.



INDIANA DEPARTMENT OF HIGHWAYS
DIVISION OF MAINTENANCE

PERFORMANCE STANDARD

ACTIVITY	Sealing Cracks	CODE	207 PM

DESCRIPTION AND PURPOSE Cleaning and sealing open cracks and joints in bituminous and concrete roadways and paved shoulder surfaces to prevent the entry of moisture and debris which leads to surface and base failure. This activity also includes sealing short sections or isolated areas of alligatored, raveled, or spalled bituminous surfaces to prevent entry of moisture and further deterioration of the surface.

AUTHORIZED BY	Subdistrict	WORK CONTROL CATEGORY	Limited

SCHEDULING Perform on areas where there is loss of seal or cracking or the joint filler is broken, brittle or missing and allowing entry of water and foreign material. This work should be scheduled in the cooler months when contraction has opened the crack or joint. Do not cover painted lines or messages without prior approval of District Traffic.

CREW SIZE 11 MEN

WORK ASSIGNMENT	QTY.
Supervisor	1
Flagman	2
Pickup or Tractor Operator	2
Air Compressor Operator	1
Tar Kettle Spray Operator	1
Laborer	2
Truck Driver/Laborer	2

EQUIPMENT	QTY.
Pickup or Tractor/Air Compressor	1
Pickup or Tractor/Tar Kettle	1
Dump Truck	2
Pickup Truck	1
Pickup/Crew Cab	1

MATERIALS

Bituminous Material
Cover Aggregate

AVERAGE DAILY PRODUCTION 2 - 4 Lane Miles

WORK METHOD

1. Place signs and other safety devices.
2. *Clean crack as required.
3. Apply bituminous material to cracks.
4. Squeegee material to force into cracks and surface voids.
5. Remove any surplus material.
6. Dust the area lightly with cover aggregate.
7. Remove signs and safety devices.

* When routing of the joint or crack on concrete surfaces is required before sealing, see Activity 219, Other Roadway and Shoulder Maintenance.

APPROVED BY:

K M Mellinger
CHIEF, DIVISION OF MAINTENANCE

D. W. Lucas
DEPUTY DIRECTOR HIGHWAY OPERATIONS

EFFECTIVE DATE JULY 1, 1982

FORM NO. MM-309(M)

Figure 2.1. Example of a Performance Standard

From the estimated need, a quantity standard (QS) is determined for each activity in each subdistrict. A QS is the amount of a given activity required per inventory unit for that activity in a particular subdistrict. For example, activity 214, joint and bump burning, is correlated with the inventory feature of lane-miles of bituminous over concrete pavement, while its unit of measure is number of bumps removed. Thus, its QS would estimate the number of bumps removed per lane-mile of bituminous over concrete pavement in a subdistrict.

The quantity standards for each activity in each subdistrict are used along with the physical feature inventory and resource costs to determine the annual work quantity of each activity for the fiscal year. This program is then reviewed by maintenance managers to determine if it is acceptable with respect to the amount and type of work the subdistricts want to program. It is also checked to insure that the estimated resource needs are not greater than the available resources. If the program is not acceptable, more visits are made to the subdistricts, and estimated needs are revised so that the amount and type of work programmed is acceptable to both subdistrict and central office managers, and that required resources are not greater than those available. This is an iterative process, and several cycles may be required before a balance is achieved between estimated needs and available resources.

Once the annual work quantity for each activity in each subdistrict has been determined, it is divided by the average daily production to determine the number of planned crew days for each activity planned. These figures are multiplied by the standard crew sizes to determine the number of man days required. Then the work calendar is determined by distributing the work load so as to keep the permanent work force busy throughout the year. Equipment and material schedules are then compiled. From estimated monthly material needs, ordering deadlines and cost schedules are established which assist the district and subdistrict personnel in scheduling and purchasing materials.

Also from the work calendar, one crew day card is printed for each crew day of work planned for each activity. A crew day card authorizes one day's work on an activity and is used to assign·men and equipment to that activity, and to report the amount of work done that day. Information recorded on a crew day card includes:

1. Activity number and name.
2. Management unit (subdistrict) number and name.
3. Road class (Interstate or Other State Highway).
4. Date.
5. Location of work.

6. Crew size and corresponding labor hours.

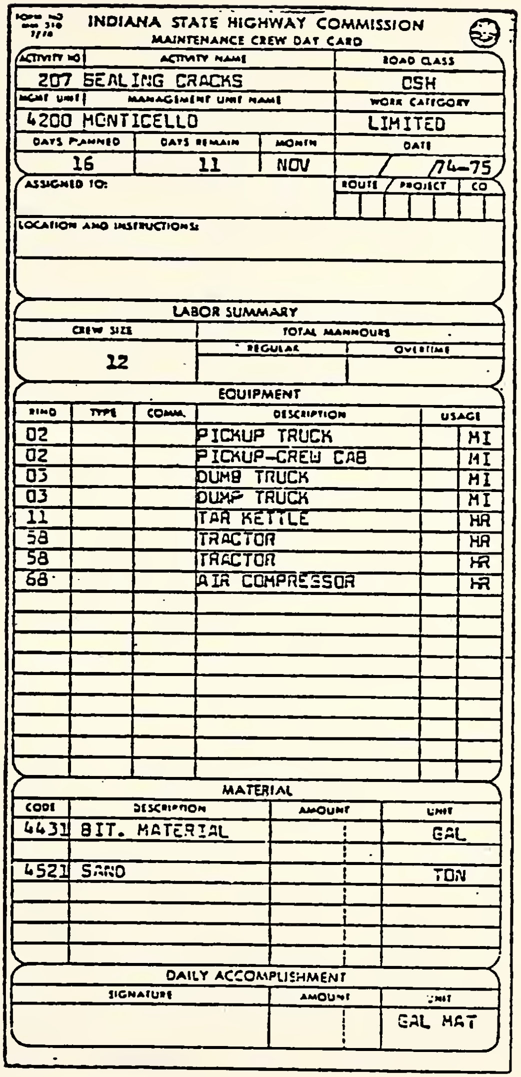
7. Equipment used and amount of use in miles or hours.

8. Materials and amounts used.

9. Total accomplishment (production unit).

Figure 2.2 is an example of a crew day card.

With the exception of equipment use, the information from the crew day card is coded and recorded on magnetic tape, and periodic reports are issued comparing factors such as actual to planned work accomplished, standard to actual crew size, and standard to actual labor productivity. However, these reports are very detailed, and there is no relatively easy way to compare the performance of the subdistricts.



[illegible]

INDIANA STATE HIGHWAY COMMISSION

MAINTENANCE CREW DAY CARD

ACTIVITY NO	ACTIVITY NAME		ROAD CLASS
207	SEALING CRACKS		OSH
MGMT UNIT	MANAGEMENT UNIT NAME		WORK CATEGORY
4200	MONTICELLO		LIMITED
DAYS PLANNED	DAYS REMAIN	MONTH	DATE
16	11	NOV	/ /74-75

ASSIGNED TO:

ROUTE	PROJECT	CO

LOCATION AND INSTRUCTIONS:

LABOR SUMMARY

CREW SIZE	TOTAL MANHOURS	
	REGULAR	OVERTIME
12		

EQUIPMENT

KIND	TYPE	COMM.	DESCRIPTION	USAGE	
02			PICKUP TRUCK		MI
02			PICKUP-CREW CAB		MI
03			DUMB TRUCK		MI
03			DUMP TRUCK		MI
11			TAR KETTLE		HR
58			TRACTOR		HR
58			TRACTOR		HR
68			AIR COMPRESSOR		HR

MATERIAL

CODE	DESCRIPTION	AMOUNT	UNIT
4431	BIT. MATERIAL		GAL
4521	SAND		TON

DAILY ACCOMPLISHMENT

SIGNATURE	AMOUNT	UNIT
		GAL MAT

Figure 2.2. Example of a Crew Day Card

CHAPTER 3

DISTRIBUTION OF MAINTENANCE EXPENDITURES

Using crew day card records for the fiscal year ending in June 1982, the distribution of maintenance expenditures by roadway class and maintenance activity type, and by subdistrict was determined. This analysis was conducted in order to determine the portion of the total maintenance expenditure due to each activity, and guide more detailed analysis to the activities which have the largest share of the maintenance program.

An analysis was also made of the relation between maintenance expenditures and the highway system physical inventory in order to identify contributing factors that can account for the variation in expenditures and productivity.

The IDOH maintenance management information system identifies sixty-three maintenance activities in eight broad categories: roadway and shoulder, roadside, drainage, bridge, traffic, winter and emergency, service, and other. Fifty-seven of these activities were included in the study and are listed in Table 3.1. It should be noted that snow and ice removal is not included in this analysis, as this

Table 3.1. Activities Included in the Study

Number	Activity Name	Unit of Measure
Roadway and Shoulder		
201	Shallow Patching	Tons of mix
202	Deep Patching	Tons of mix
203	Premix Leveling	Tons of mix
204	Full Width Shoulder Seal	Foot miles
205	Seal Coating	Lane Miles
206	Sealing Longitudnal Cracks and Joints	Linear Miles
207	Sealing Cracks	Lane Miles
209	Cutting Relief Joints	Linear Feet
210	Spot Repair of Unpaved Shoulders	Tons of Aggregate
211	Blading Shoulders	Shoulder Miles
212	Clipping Unpaved Shoulders	Shoulder miles
213	Recondition Unpaved Shoulders	Shoulder miles
214	Joint and Bump Burning	Bumps Removed
219	Other Roadway & Shoulder	Manhours
Roadside		
221	Machine Mowing	Swath miles per mower
222	Brush Cutting	Manhours
223	Herbicide Treatment	Manhours
224	Seeding and/or Fertilizing	Manhours
225	Topping, Trimming, Removal of Large Trees	Trees
226	Stump Removal	Stumps
227	Spot Mowing and Hand Trimming	Manhours
228	Right-of-Way Fence Repair	Linear feet
229	Other Roadside	Manhours

Table 3.1. Continued

Number	Activity Name	Unit of Measure
Drainage		
231	Clean and Reshape Ditches	Linear feet
232	Inspect Minor Drainage Structures	Structures
233	Pipe Replaçement	Location
234	Motor Patrol Ditching	Ditch Mile
235	Cleaing Minor Drainage Structures	Structures
239	Other Drainage	Manhours
Bridge		
241	Hand Cleaning Bridges	Decks
243	Bridge Repair	Manhours
244	Flushing Bridges	Decks
245	Patching Bridge Decks	Square feet
249	Other Bridge	Manhours
Traffic		
251	Subdistrict Sign Maintenance	Manhours
252	District Sign Maintenance	Manhours
253	District Sign Replacement	Signs
254	District Signal Maintenance	Manhours
255	Paint Centerlines	Centerline miles
256	Paint Edgelines	Edgeline Miles
257	Special Markings/Paint Pvmt Messages	Manhours
258	Guardrail Maintenance	Linear feet
259	Other Traffic	Manhours

Table 3.1. Continued

Number	Activity Name	Unit of Measure
Winter and Emergency		
261	Emergency Maintenance	Manhours
265	Stockpiling Winter Materials	Manhours
266	Winter Night Patrol	Manhours
269	Other Winter	Manhours
Service		
272	Roadside Park, Rest Area, Weigh Station Maint	Manhours
273	Work for Dept. Natural Resources	Manhours
274	Work for State Institutions	Manhours
275	Full Width Litter Pickup	ROW Pass miles
276	Spot Litter Pickup	Manhours
277	Roadway Cleaning	Manhours
279	Other Service	Manhours
Other		
284	Materials Handling and Storage	Manhours
287	Detour Maintenance	Manhours
289	Other Support Activities	Manhours

activity does not follow any trend, and it depends entirely on weather. Since the crew day card records include only labor and material used, equipment depreciation and fuel costs are not considered. Along with the crew day card data, information on hourly wage rates and average unit material costs for the various activities was used to generate the expenditure data. Table 3.2 lists the hourly wage rates used for each activity, and Table 3.3 lists the unit material costs.

Maintenance Expenditure By Roadway Class and Activity Type

Table 3.4 presents a breakdown of routine maintenance expenditure by activity type and roadway class. Approximately $24.9 million was spent in fiscal year 1981-82, with 11.5 percent going to the Interstate system and 88.5 percent to the Other State Highways. Roadway and shoulder activities account for the highest share of maintenance costs at 38 percent, with traffic activities second at 25 percent. Roadside activities were third at just over 10 percent, and drainage activities fourth at just under 10 percent.

An analysis of expenditure types by individual activity reveals that activity 201, shallow patching, was the largest single activity at 31.5 percent of roadway items, and 12.1 percent of total reported maintenance expenditure. Activity 256, painting edgelines, placed second at 37.7 percent of

Table 3.2. Hourly Wage Rates for Activities in Study[1]

ACTIVITY NUMBER(2)	AVERAGE LABOR WAGE ($/HR)	ACTIVITY NUMBER(2)	AVERAGE LABOR WAGE ($/HR)
201	5.81	241	5.91(3)
202	5.97	243	5.99
203	6.04	244	5.91(3)
205	6.21(3)	249	5.91(3)
206	6.22	251	6.09
207	5.81	252	6.09(3)
209	6.07	253	6.09(3)
210	5.81	254	6.09(3)
211	5.95	255	6.09(3)
212	6.16	256	6.09(3)
213	6.09	257	6.00
214	6.01	258	5.88
219	5.85	259	5.92
221	5.88	261	6.09
222	5.99	265	5.76
223	5.95	266	5.76(3)
224	5.95	269	5.84
225	6.13	272	5.67
226	5.95	273	5.89(3)
227	5.95	274	5.89
228	5.88	275	5.67
229	5.92	276	5.67
231	5.97	277	5.88
232	6.09	279	5.67
233	5.97	284	5.95
234	6.11	287	6.10
235	5.97	289	6.09
239	5.88		

1 -- Source: Reference 13.
2 -- Refer to Table 3.1 for Activity Names
3 -- Value not reported in Reference 13. Wage rate is assumed based on similar activities.

Table 3.3. Selected Average Material Costs[1]

CODE NUMBER	DESCRIPTION	UNIT COST ($)	UNIT
4061	Fuel Oil	1.05	Gallon
4121	Grass Seed	0.85	Pound
4122	Fertilizer	0.07	Pound
4221	Culvert Pipe	14.00	Linear Foot
4222	Underdrain Pipe	14.00	Linear Foot
4241	Bridge Paint	12.00	Gallon
4242	Primer	10.00	Gallon
4244	Yellow Quick Dry Paint	4.52	Gallon
4245	Glass Beads	0.21	Pound
4247	Solvent	1.65	Gallon
4248	Greenlite Powder	1.32	Pound
4251	Aggregate	4.10	Ton
4252	Seal/Cover Aggregate	3.00	Ton
4253	Rip Rap	4.32	Ton
4254	Silica Sand	65.00	Ton
4371	Herbicides	22.00	Gallon
4372	Dry Herbicides	1.14	Pound
4401	Preformed Joint Filler	3.14	Linear Foot
4411	Guardrail	6.67	Linear Foot
4421	Fence	1.16	Linear Foot
4431	Bituminous Material	0.78	Gallon
4441	Bituminous Mixture, Hot	25.50	Ton
4442	Bituminous Mixture, Cold	25.50	Ton
4451	Readymix Concrete	39.35	Cubic Yard
4452	Cement	4.50	Bag
4453	Sacrete	0.51	Pound
4461	Signs	18.00	Each
4462	Sign Posts	7.80	Each
4463	Delineators	0.29	Each
4464	Delineator Posts	3.00	Each
4486	White Quick Dry Paint	4.02	Gallon
4501	Salt	28.00	Ton
4511	Calcium Chloride	0.07	Pound
4521	Abrasives	2.40	Ton

[1] -- Source: Reference 5.

Table 3.4. 1981-82 Routine Maintenance Expenditure by Activity Type and Highway Class (Dollars)

Activity Class	Interstate	Other State Highway	Total	Percent of Total
Roadway & Shoulder	1,048,718	8,490,598	9,539,322	38.3
Roadside	328,440	2,260,381	2,588,822	10.4
Drainage	149,173	2,253,835	2,403,008	9.6
Bridge	174,454	433,227	607,682	2.4
Traffic	840,758	5,488,792	6,329,548	25.4
Winter & Emergency	70,470	1,200,431	1,270,902	5.1
Service	253,846	971,536	1,225,382	4.9
Other	14,114	963,282	977,396	3.9
Total	2,879,973	22,062,082	24,942,062	100.0

Table 3.5. 1981-82 Expenditures for Roadway & Shoulder and Traffic Activities (Dollars)

ACT	INTERSTATE	OTHER ST HWY	TOTAL	% TOTAL EXP
Roadway & Shoulder				
201	248,668	2,757,152	3,005,820	12.1
202	63,628	426,254	489,883	2.0
203	27,798	371,294	399,092	1.6
204	393,455	443,255	836,711	3.4
205	0	1,131,675	1,131,675	4.5
206	79,466	246,402	325,868	1.3
207	114,387	1,498,794	1,613,181	6.5
209	15,911	89,570	105,481	0.4
210	14,042	612,339	626,381	2.5
211	23,966	168,619	192,584	0.8
212	22,139	277,584	299,723	1.2
213	0	87,541	87,541	0.4
214	10,207	166,690	176,896	0.7
219	35,051	213,432	248,483	1.0
Traffic				
251	134,535	1,102,521	1,237,056	5.0
252	53,782	38,076	91,858	0.4
253	38,383	653,081	691,463	2.8
254	0	0	0	0.0
255	81,407	1,009,092	1,090,499	4.4
256	361,483	2,027,036	2,388,519	9.6
257	32,921	370,237	403,158	1.6
258	125,111	170,724	295,835	1.2
259	13,136	118,023	131,160	0.5

traffic items and 9.6 percent overall. Ranked third overall was activity 207, sealing cracks, at 6.5 percent of total expenditure, followed by activity 251, subdistrict sign maintenance, at 5.0 percent. Table 3.5 presents the total breakdown by activity for roadway and traffic items.

Maintenance Expenditure by Subdistrict and Activity Type

Since this study is primarily concerned with providing a method for comparing subdistricts, an analysis of expenditure by subdistrict was conducted. In order to compare each subdistrict on a comparable basis, maintenance expenditure was plotted against lane-miles for each subdistrict. In general, one would expect the subdistricts with more roadways to require more maintenance activity. Figures 3.1, 3.2, and 3.3 show total expenditure vs. lane-miles for the Interstate, Other State Highway, and Total highway systems, respectively. The expenditure data in this analysis do not include activities that are performed by district-wide crews. These activities are district sign maintenance, district sign replacement, district signal maintenance, painting centerlines, painting edgelines, and portions of special markings/painting pavement messages and other traffic control.

While there is a considerable variation, in general, the subdistricts with higher lane-miles spent more money on maintenance. A linear regression analysis was conducted for

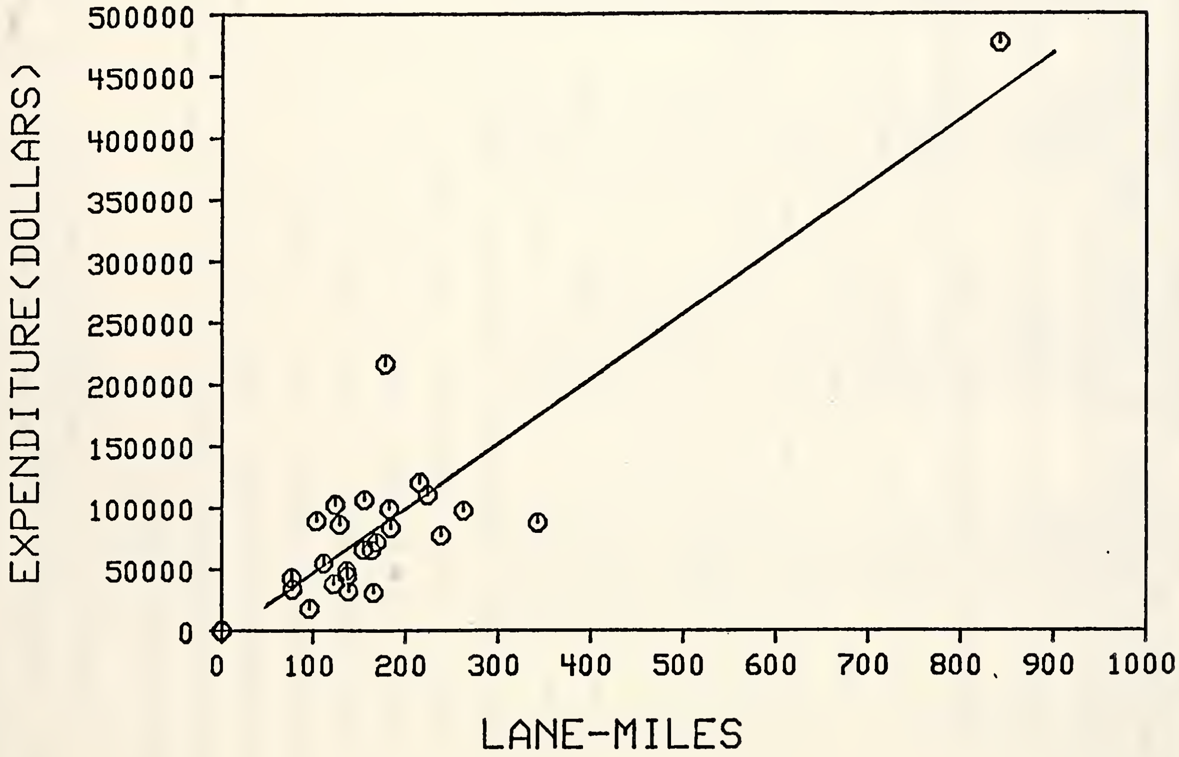


Figure 3.1. Expenditure vs. Lane-Miles for All Activites FY 1982-83 (Interstate System)

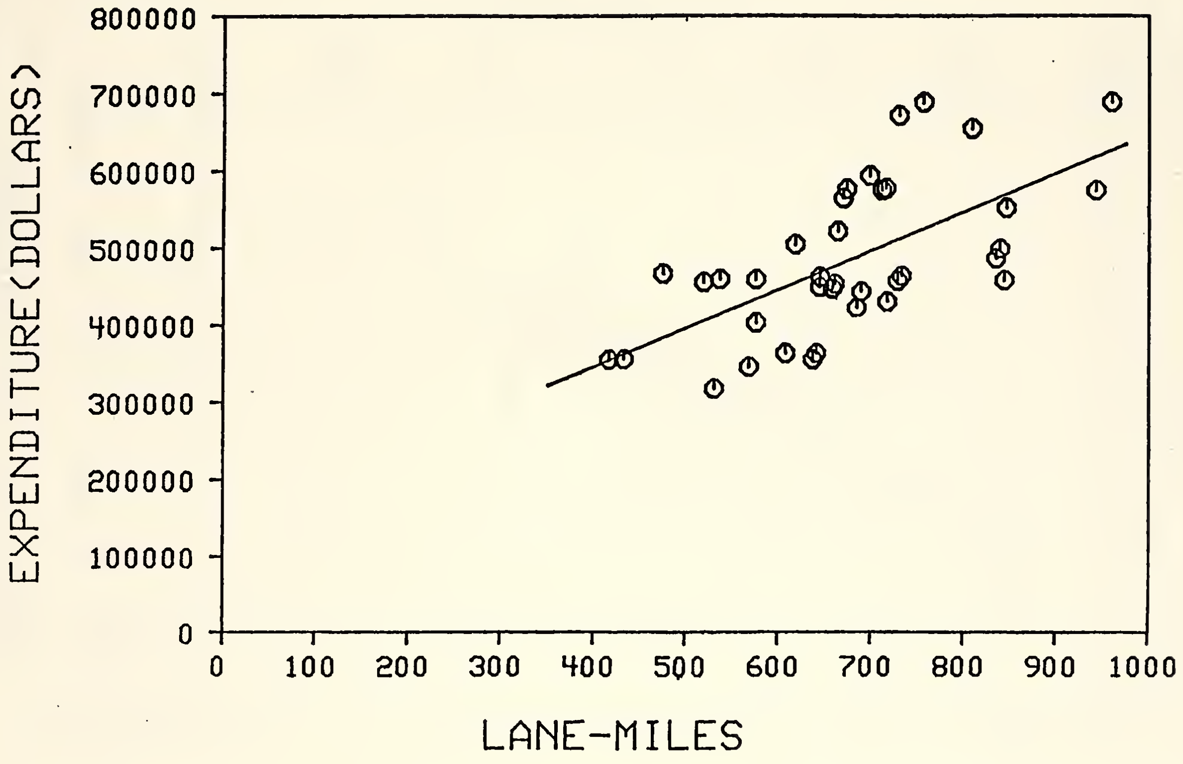


Figure 3.2. Expenditure vs. Lane-Miles for All Activites FY 1982-83 (Other State Highway System)

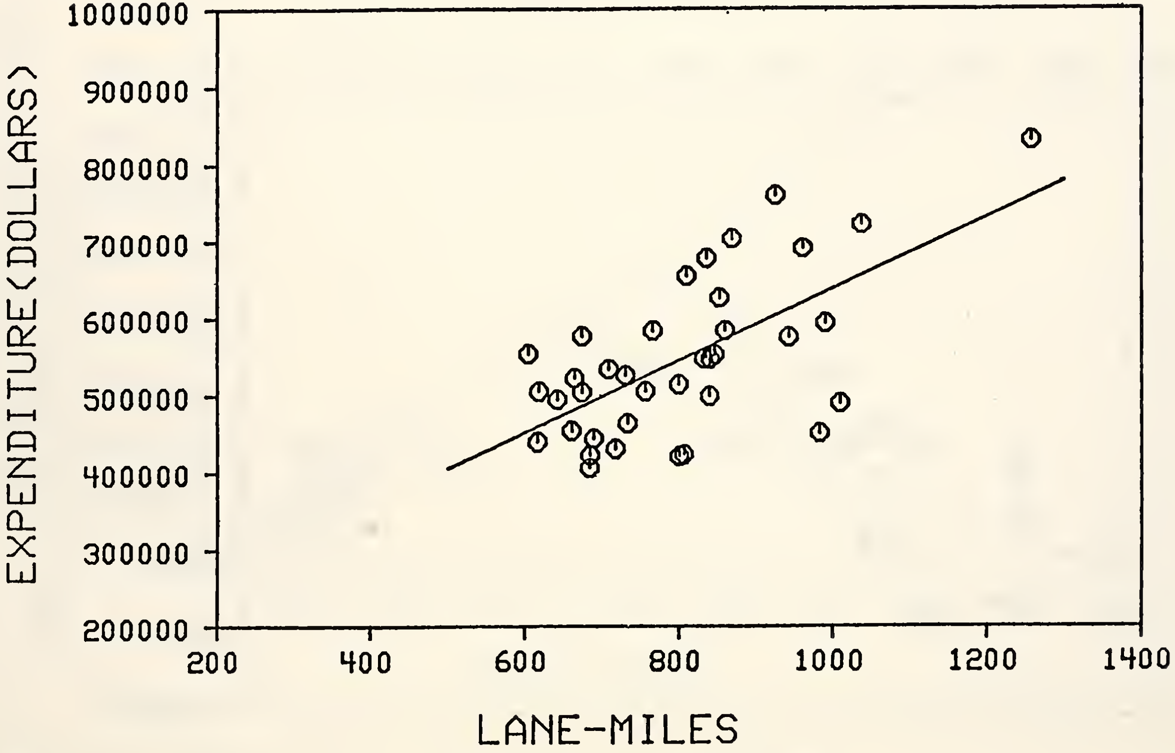


Figure 3.3. Expenditure vs. Lane-Miles for All Activites FY 1982-83 (Total Highway System)

each of the cases. The Interstate case yielded a r^2 value of 0.85, while the Other State Highway case had a r^2 value of 0.41, and the Total system case had a r^2 value of 0.41.

In addition to considering total expenditure, expenditure for each activity class was plotted against an appropriate physical feature. Roadside expenditures were plotted against the number of right-of-way pass-miles, (A pass-mile is one pass along the right-of-way for a distance of one mile.) while ′expenditures for drainage activities were plotted against the number of road-miles, and bridge expenditures were plotted versus number of bridges. Expenditures for the other activity types were plotted against lane-miles.

Figure 3.4 shows expenditure for roadway and shoulder activities versus lane-miles for the total highway system. Again, in general, expenditure increases with increasing numbers of lane-miles, but a linear regression analysis yielded a r^2 value of .15 for the Total system, indicating that the total expenditure for roadway and shoulder activities cannot be predicted on the basis of lane-miles.

Table 3.6 summarizes the r^2 values obtained for the various activity types and roadway classes. With the exception of roadway and shoulder activities, the r^2 values for the Interstate system are higher than those for the Other State Highway system. This result is not surprising

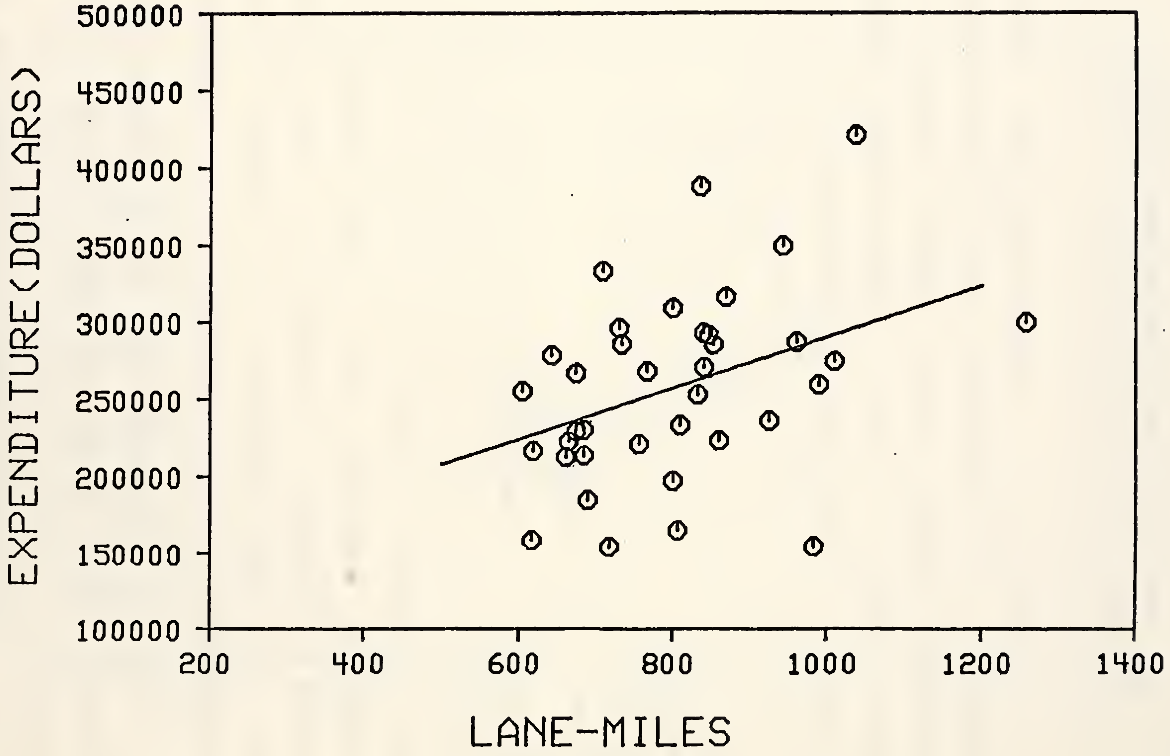


Figure 3.4. Expenditure vs. Lane-Miles for Roadway and Shoulder Activities FY 1982-83 (Total Highway System)

Table 3.6. R^2 Values for 1981-82 Expenditures

ACTIVITY CLASS	INDEPENDENT VARIABLE	R SQUARED		
		INT	OSH	TOT
Rdwy & Shldr	Lane-miles	.43	.44	.15
Roadside	ROW Pass-miles	.34	.13	.11
Drainage	Road-miles	.61	.30	.16
Bridge	Bridges	.86	.30	.16
Traffic	Lane-miles	.79	.00	.15
Winter & Emgy	Lane-miles	.75	.04	.11
Service	Lane-miles	.85	.04	.05
Other	Lane-miles	.39	.00	.03
Total	Lane-miles	.85	.41	.41

INT -- Interstate
OSH -- Other State Highways
TOT -- Total Highway System

considering the differences in the two systems. The Interstate system is homogeneous in terms of the type of highway facility. All interstates are, by definition, grade separated, high-speed, multilane highways. The Other State Highway system is heterogeneous, consisting of all state highways other than Interstates. Everything from a multilane divided freeway, to a rural two-lane highway, to an urban street can be found in the Other State Highway system. Maintenance requirements would be expected to be relatively similar throughout a homogeneous group, such as the Interstate system. On the other hand, requirements would be expected to vary widely throughout the heterogeneous Other State Highway system, depending on the particular type of highway being considered. It is also reasonable to expect that heavily travelled highways require more maintenance than those with low traffic volumes. In general, Interstate highways are heavily travelled, while highways in the Other State Highway system range from high to low traffic volumes. Thus, it is reasonable that maintenance expenditure could be better predicted on the basis of total lane-miles for the Interstate system, than for the Other State Highway system.

Examining the values for each activity class, all Interstate activities have relatively high values except for roadside, and other. The roadside class consists mainly of grass and weed cutting, brush clearing, tree trimming, and so on. Perhaps these activities have a lower priority than the others, which have a more pronounced effect on the highway user, such as roadway, bridge, drainage, and traffic activities. Thus, the more important activities items may be taken care of first, with the amount of roadside work depending on the time and resources remaining. The "other" activity class includes items such as materials handling and detour maintenance. Items such as detour maintenance would not be expected to be related to lane-miles, since it depends on construction work in progress.

Considering the Other State Highway category, three activity types have relatively high values, roadway and shoulder, drainage, and bridge, while the others have r^2 values near zero. This may indicate that all highways require at least some maintenance for the major categories of pavement (roadway), drainage, and bridges, while need for the other activities varies more widely due to the heterogeneous character of the OSH system. For example, rural two-lane highways and urban streets with narrow or no grassy rights-of-way would require little or no roadside mowing and litter pickup, activities that fall into the roadside and service categories. Normally, the traffic category would be expected to have a higher r^2, since all roadways must have painted center and edgelines. However, these activities are performed by district-wide crews, and are not included in this analysis. The remaining activities are mostly sign and guardrail maintenance, the need for which would vary widely with the type of highway.

Considering the Total highway system, the r^2 values for each particular activity class are rather low. This is not surprising considering that the Total system is also heterogeneous in highway types, since it contains all the state highways. The low values indicate the the expenditure for each activity class cannot be predicted based on the number of lane-miles (or road-miles, bridges, or right-of-way pass-miles when appropriate) in a subdistrict. The number of lane-miles is a better predictor of expenditure for all activities combined, with a r^2 of 0.41.

Figure 3.5 is a plot of expenditure per lane-mile versus lane-miles for all activities on the Total highway system. There is a general trend of decreasing expenditure per lane-mile with increasing total lane-miles. This, along with the general trend of increasing total expenditure with increasing total lane-miles, can be interpreted as discussed below.

If it is assumed that the amount of programmed maintenance and maintenance expenditures are based on need, and that each subdistrict maintains its highways at about the same level of quality, the decreasing expenditure per lane-mile could indicate that there are economies of scale involved in routine maintenance operations. Thus, a

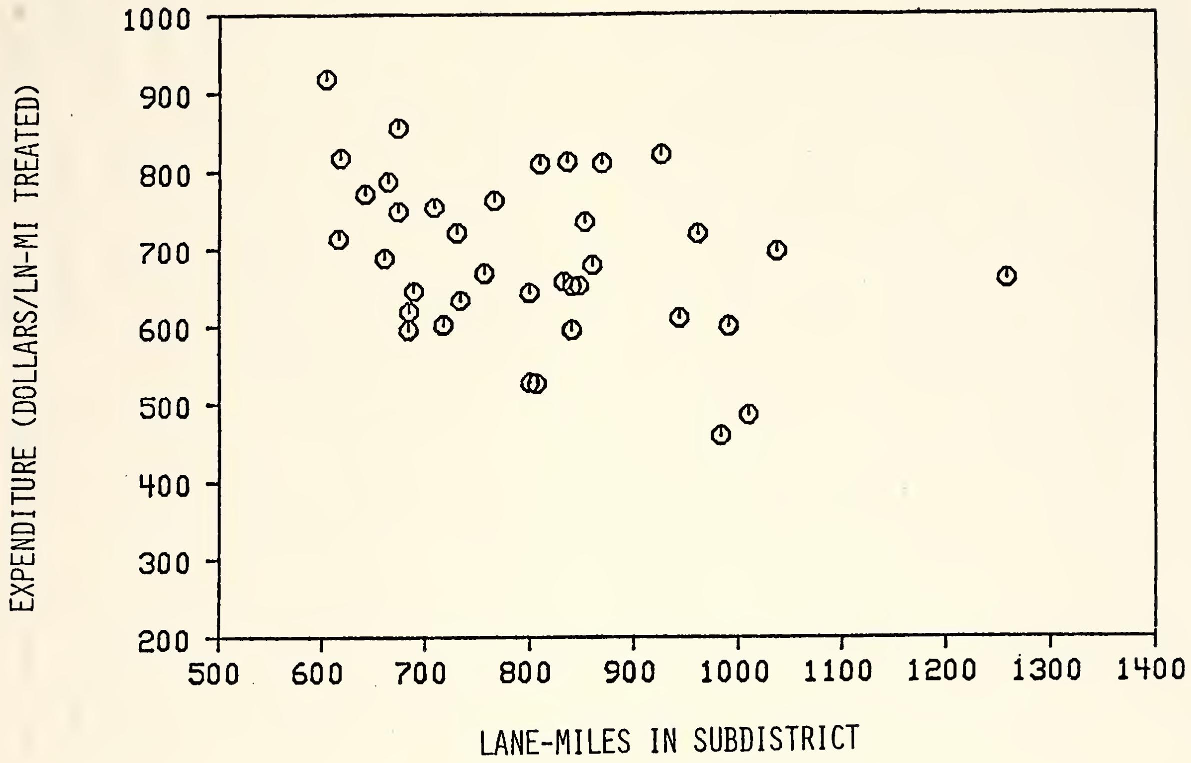


Figure 3.5. Expenditure per Lane-Mile vs. Lane-Miles for All Activities FY 1982-83 (Total Highway System)

subdistrict with more lane-miles does more total work, and the scheduling of personnel and equipment is more efficient, resulting in a lower cost per lane-mile. Subdistricts with more lane-miles typically also have more multi-lane highways and the multi-lane highways require proportionately less maintenance per lane-mile for features such as shoulders, ditches, and roadside maintenance. Also, the multi-lane highways receive more rehabilitation funds from other funding sources, thereby relieving the pressure on routine maintenance. However, if the assumption of equal quality levels of maintenance is dropped, the decreasing cost per lane-mile may also indicate that the larger subdistricts, having more territory to cover, perhaps spend less time on a given activity and perform the work at a lower quality level.

If one assumes that there exists an "average need" for highway maintenance throughout the state, it would be expected that subdistricts with more highways would have higher total needs and thus higher total expenditures. But the expenditure per lane-mile would be expected to remain relatively constant. Since expenditure per lane-mile does tend to decrease with increasing total lane-miles, perhaps the increase in total expenditure is not adequate to meet the needs. In this case, the results would lead to the conclusion that maintenance expenditures, or the amount of maintenance programmed in a subdistrict, is not determined solely on the basis of need.

It seems likely that a combination of the factors discussed may be responsible for the observed trends. However, the total amount of money allocated currently to a subdistrict is not strictly based on formally-determined needs; rather, the resources available are distributed among districts and subdistricts through a process involving a series of meetings among Subdistrict, District and Central Office personnel. Research in the area of assessment of actual routine maintenance needs will be of much help in programming routine maintenance. However, that subject is not within the scope of the current study.

CHAPTER 4

EVALUATION OF SUBDISTRICT PRODUCTIVITY

By using the maintenance management information system to monitor the productivity of the subdistricts in executing the maintenance program, managers can spot problem areas where resources may not be being used as efficiently as possible. On the other hand, bright spots may be discovered where subdistrict personnel have developed highly efficient methods of accomplishing their maintenance program. This can lead to corrective action being taken in the former case, and dissemination of methods and recognition of those responsible in the latter, resulting in an improvement in overall efficiency of the state's maintenance forces.

A procedure for monitoring subdistrict productivity, using elements of the IDOH Maintenance Management Information System already in place, was prepared. The procedure is based on computer and manual analysis of crew day card records and field observations of maintenance crews at work.

Description of Computer Program

A computer program was developed to use the crew day card data to produce relatively simple and straight-forward reports showing various factors by which subdistrict performance may be assessed.

On the basis of the crew day card records, the program determines the number of times a given activity was performed by each subdistrict, the total amount of work accomplished in the time period under study, the average accomplishment per crew day, the average crew size, the average number of manhours (both regular and overtime) per crew day, and the number of manhours per production unit. Also determined are the percent of the time a given material is used, the average quantity of the material when it is used, and the average quantity of material per production unit, when that material is used. The average cost per production unit is calculated, along with the labor cost and material cost per production unit. A summary of production amounts, labor and material use for each of the six districts and the state as a whole are also calculated. The subdistrict summary information can also be presented in bar chart form.

After determining these values for each subdistrict, the program takes the average cost figures for each subdistrict and calculates the average and standard deviation. Then, the average cost for each subdistrict is checked to see if it falls outside the range of the average plus or minus a given number of standard deviations. These deviate units are then listed.

Figure 4.1 is an example of the first page of output from the analysis of activity 201, shallow patching for the period from July 1982 through June 1983. This page presents all the input parameters necessary for the analysis. As it can be seen, the production unit of activity 201 is measured in terms of tons of bituminous mixture placed. There are six materials specified for this activity indicating that up to six of these materials may be used. These are determined by referring to the appropriate performance standard. The input data specify the code number and material description. For example, hot bituminous mixture is coded onto crew day cards as material 4441. The next column contains the material′s unit price in dollars per unit of measure. The last column contains the maximum expected quantity of this material, a value used to detect typographical errors in entering the crew day card data. If a quantity of 30.0 tons of mix or greater were read, we would suspect that this is a coding error. That record would be rejected, and a data check message would be printed.

Next, the wage rates for regular and overtime hours are listed. These values are used to calculate the labor cost. Maximum expected values for crew size, manhours, and work accomplishment are then listed. These are used to check for typographical errors as explained before. The beginning and ending dates of the analysis period are listed. The shortest time period that can be considered is one month.

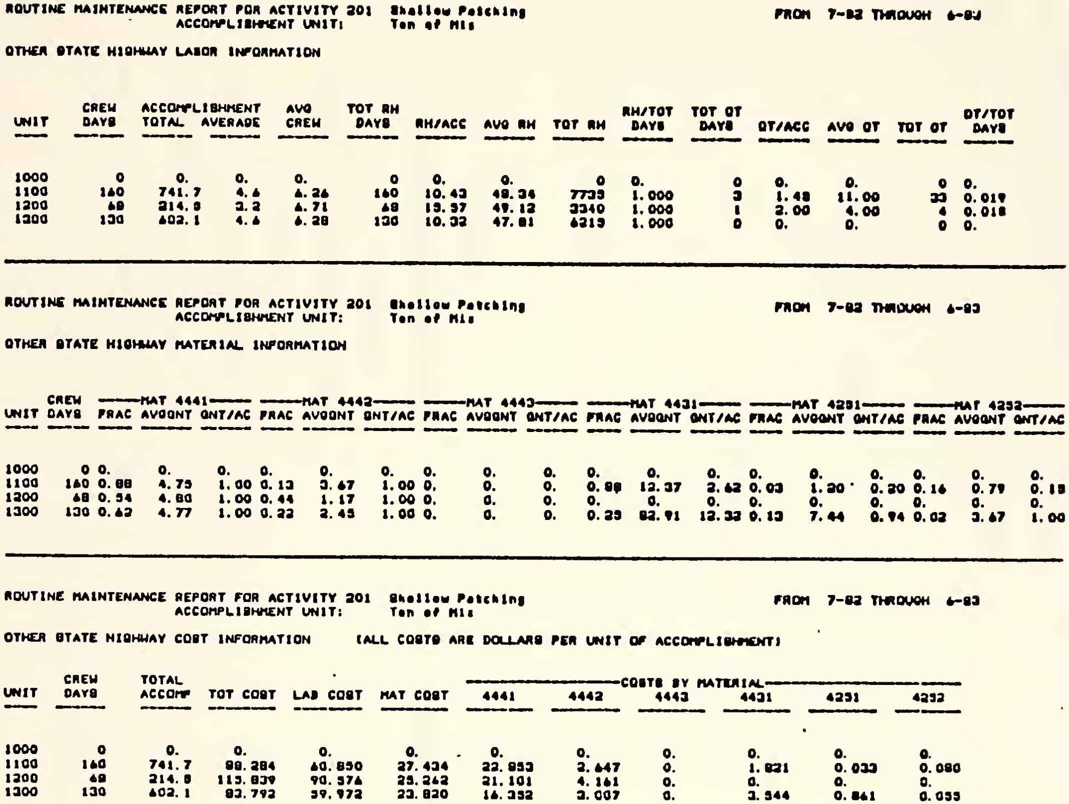
```
ROUTINE MAINTENANCE REPORT


INPUT PARAMETERS


ACTIVITY   201   Shallow Patching                              ACCOMPLISHMENT UNIT:   Ton of Mix

THE  6 MATERIALS SPECIFIED FOR THIS ACTIVITY ARE:

     CODE    DESCRIPTION                                    UNIT COST($)  UNIT          MAX EXPECTED QUANTITY
     ----    -----------                                    ------------  ----          ---------------------
     4441    Bituminous Mixture HOT                               25.50   Ton                           30.00
     4442    Bituminous Mixture COLD                              25.50   Ton                           30.00
     4443    Salvage Bituminous Mixture                           0.      Ton                           30.00
     4431    Bituminous Material                                  0.78    Gallon                      5000.00
     4251    Aggregate                                            4.10    Ton                          500.00
     4252    Seal/Cover Aggregate                                 3.00    Ton                          500.00

LABOR COST:
     Regular Hour     5.81
     Overtime Hour    5.81

VALUES FOR CHECKING DATA:
    Max Crew              30
    Max Hours            240
    Max Production      30.0


ANALYSIS PERIOD:     7-82    through     6-83

ANALYSIS FOR:
               OTHER STATE HIGHWAY

DEVIATIONS WILL BE DÈTECTED USING COST PRODUCTIVITY BEYOND + OR - 1.000 STANDARD DEVIATION(S)

PRINT CHART FOR:
                AVERAGE COST PER ACCOMPLISHMENT UNIT
                LABOR HOURS PER ACCOMPLISHMENT UNIT
                TOTAL PERIOD ACCOMPLISHMENT
                AVERAGE DAILY ACCOMPLISHMENT
                AVERAGE CREW SIZE
                QUANTITY OF MATERIAL 4431 PER ACCOMPLISHMENT UNIT


Beginning Record=        1
```

Figure 4.1. Example of Input Summary Page

The next few lines indicate which program options have been selected. The program can analyze maintenance work for the Interstate system, for the Other State Highway system, or the Total highway system. The next line indicates that a subdistrict will be identified as being deviate in productivity based on average cost per unit of accomplishment plus or minus one standard deviation. The number of standard deviations to be used is entered by the analyst. Finally, the types of bar charts that will be printed are listed. There are six charts that may be printed for each highway class.

As mentioned earlier, there is a series of data check messages issued by the program. In addition to the previously described messages, the program also prints a message if it encounters a material that was not specified in the input information. An examination of the data records indicates that most of these messages result from coding errors. The number of rejected data records is very small. Considering the 1982-83 data for shallow patching, a total of 9 out of approximately 12070, records were rejected, or 0.07 percent.

Figure 4.2 shows a part of the labor summary page for the Other State Highway system. The first column indicates the management unit, or subdistrict. Consider unit 1200, the Crawfordsville subdistrict. The "CREW DAYS" column



```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                                      FROM  7-82 THROUGH  6-83
                      ACCOMPLISHMENT UNIT:       Ton of Mix

OTHER STATE HIGHWAY LABOR INFORMATION


        CREW    ACCOMPLISHMENT    AVG      TOT RH                                RH/TOT   TOT OT                                       OT/TOT
UNIT    DAYS    TOTAL  AVERAGE    CREW      DAYS    RH/ACC   AVG RH   TOT RH      DAYS     DAYS     OT/ACC    AVG OT    TOT OT       DAYS
----   ------   -----  -------   ------    ------   ------   ------   ------     ------   ------    ------    ------    ------      ------

1000       0      0.       0.       0.          0    0.       0.           0     0.             0    0.        0.             0    0.
1100     160    741.7      4.6      6.26      160   10.43    48.34      7733     1.000          3    1.48     11.00          33    0.019
1200      68    214.8      3.2      6.71       68   15.57    49.12      3340     1.000          1    2.00      4.00           4    0.018
1300     130    602.1      4.6      6.28      130   10.32    47.81      6219     1.000          0    0.        0.             0    0.
```

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                                      FROM  7-82 THROUGH  6-83
                      ACCOMPLISHMENT UNIT:       Ton of Mix

OTHER STATE HIGHWAY MATERIAL INFORMATION


      CREW   -------MAT 4441------   -------MAT 4442------   -------MAT 4443------   -------MAT 4431------   -------MAT 4231------   -------MAT 4232------
UNIT  DAYS   FRAC  AVGQNT  QNT/AC   FRAC  AVGQNT  QNT/AC   FRAC  AVGQNT  QNT/AC   FRAC  AVGQNT  QNT/AC   FRAC  AVGQNT  QNT/AC   FRAC  AVGQNT  QNT/AC
----  ----   ----  ------  ------   ----  ------  ------   ----  ------  ------   ----  ------  ------   ----  ------  ------   ----  ------  ------

1000     0  0.      0.       0.    0.      0.       0.    0.       0.      0.    0.       0.      0.    0.       0.      0.    0.       0.      0.
1100   160  0.88    4.75     1.00  0.13    3.67     1.00  0.       0.      0.    0.98    12.37    2.62  0.03     1.20    0.20  0.16     0.79    0.18
1200    68  0.54    4.80     1.00  0.44    1.17     1.00  0.       0.      0.    0.       0.      0.    0.       0.      0.    0.       0.      0.
1300   130  0.62    4.77     1.00  0.22    2.45     1.00  0.       0.      0.    0.23    82.91   12.32  0.13     7.44    0.94  0.02     3.67    1.00
```

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                                      FROM  7-82 THROUGH  6-83
                      ACCOMPLISHMENT UNIT:       Ton of Mix

OTHER STATE HIGHWAY COST INFORMATION       (ALL COSTS ARE DOLLARS PER UNIT OF ACCOMPLISHMENT)

       CREW        TOTAL                                        --------------------COSTS BY MATERIAL------------------------
UNIT   DAYS        ACCOMP   TOT COST   LAB COST   MAT COST     4441      4442      4443      4431      4231      4232
----  ------       ------   --------   --------   --------    ------    ------    ------    ------    ------    ------

1000       0        0.        0.         0.         0.          0.        0.        0.        0.        0.        0.
1100     160      741.7     88.284     60.850     27.434      22.853     2.647     0.        1.821     0.033     0.080
1200      68      214.8    115.839     90.576     25.242      21.101     4.161     0.        0.        0.        0.
1300     130      602.1     83.792     59.972     23.820      16.352     3.007     0.        3.544     0.841     0.055
```

Figure 4.2. Excerpts from Labor, Material, and Cost Summaries

indicates that the activity shallow patching was carried out 68 times during the analysis period. The next column refers to total accomplishment of 214.5 tons of bituminous mix placed. The average accomplishment per crew day was 3.2 tons, and the average crew size was 6.7 men. The next five columns provide information on the number of regular labor hours used. "TOT RH DAYS" indicates that regular labor hours were used in all 68 instances of shallow patching. The average number of regular hours per ton placed was 15.57, the average number of regular hours per crew day was 49.12, and the total regular hours attributed to this activity during the year was 3340. The next column indicates the fraction of shallow patching crew days on which regular hours were used.

The last five columns give similar labor data for overtime used. There was only one time when overtime hours were used for shallow patching in the Crawfordsville subdistrict, and the average overtime hours per ton placed was 2.00. The average overtime per crew day, when overtime was used, was 4.00 hours, and the total number of overtime hours used for shallow patching that year was 4. The last column indicates that overtime was used 1.5 percent of the time that activity 201 was carried out.

Figure 4.2 also presents a part of the material use information. Again considering unit 1200, we see that the total number of times that activity was performed is repeated here. The next three columns give information about the use of material 4441, hot bituminous mixture, computed as shown below.

$$\text{FRAC} = \frac{\text{TR}}{\text{TR}_{4441}}$$

$$\text{AVGQNT} = \frac{\text{TQ}_{4441}}{\text{TR}_{4441}}$$

$$\text{QNT/AC} = \frac{\text{TQ}_{4441}}{\text{TAC}_{4441}}$$

Where:

FRAC = Fraction of the time material 4441 is used in activity,

TR = Total number of times activity is performed,

TR_{4441} = Total number of times material 4441 is used in activity,

AVQNT = Average quantity of material 4441 when it is used,

TQ_{4441} = Total quantity of material 4441 used,

QNT/AC = Average quantity of material per accomplishment unit,

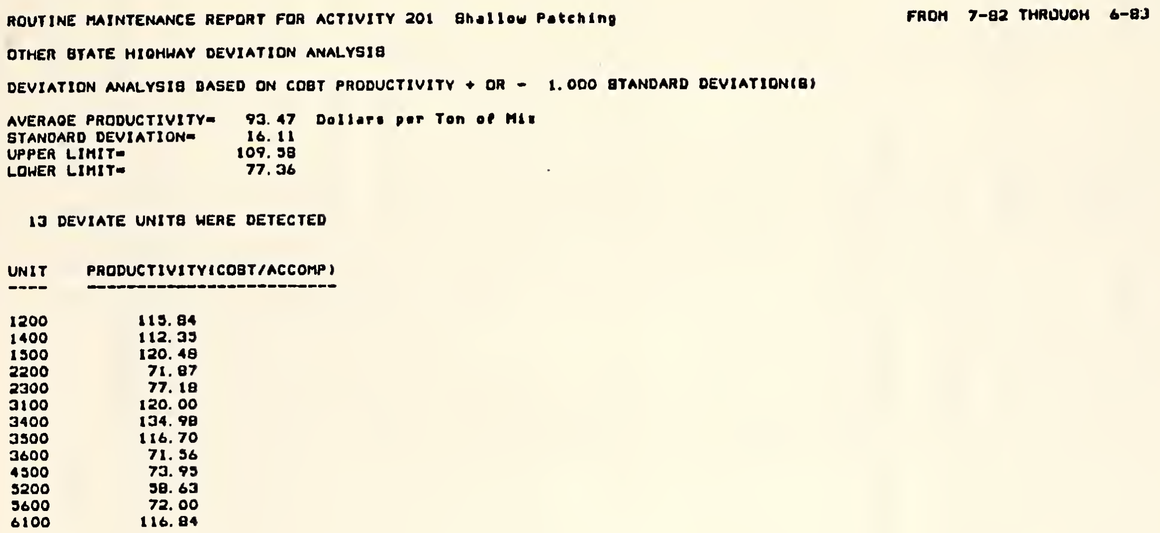
TAC_{4441} = Total accomplishment for activity when material 4441 is used.

Thus, material 4441, hot bituminous mixture, was used 54% of the time that shallow patching was carried out, and the average quantity used was 4.8 tons per day. The average amount of material 4441 per unit of accomplishment was 1.00 ton. (This particular measure for this activity will always be 1.00, because the accomplishment is measured in tons of mix placed. Thus, tons of bituminous mix used divided by tons of accomplishment will equal 1.00.) Similar calculations are made for each specified material.

The last section of Figure 4.2 presents a part of the average cost per subdistrict data. Again, the subdistrict number, the number of times the activity was performed, and the total production accomplishment are entered. The fourth column lists total cost per ton of shallow patching for materials and labor. In the case of Crawfordsville, unit 1200, the average cost for shallow patching was $115.84. The labor portion of this cost was $90.58, while the materials accounted for $25.26 per ton. The last columns break down material cost by material type.

Similar summaries are provided for each district and for the state as a whole.

Figure 4.3 indicates the results of the productivity deviation analysis. The average productivity of all 37 subdistricts was $93.47 per ton, with a standard deviation of $16.11 per ton. Using the average plus or minus one standard deviation, the lower limit is set at $77.36 per ton, and the upper limit is $109.58 per ton. All subdistricts whose average cost per ton for shallow patching falls outside this range are listed.



```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                    FROM  7-82 THROUGH  6-83

OTHER STATE HIGHWAY DEVIATION ANALYSIS

DEVIATION ANALYSIS BASED ON COST PRODUCTIVITY + OR -  1.000 STANDARD DEVIATION(S)

AVERAGE PRODUCTIVITY=    93.47  Dollars per Ton of Mix
STANDARD DEVIATION=      16.11
UPPER LIMIT=            109.58
LOWER LIMIT=             77.36

  13 DEVIATE UNITS WERE DETECTED

UNIT    PRODUCTIVITY(COST/ACCOMP)
----    -------------------------

1200          115.84
1400          112.35
1500          120.48
2200           71.87
2300           77.18
3100          120.00
3400          134.98
3500          116.70
3600           71.56
4500           73.95
5200           58.63
5600           72.00
6100          116.84
```

Figure 4.3. Example of Productivity Deviation Analysis

Figure 4.4 is a bar chart showing the average labor hours per accomplishment unit. The average and standard deviation are listed at the top. To the right, the values for hours, cost, and accomplishment are listed. Charts for the other factors are similar.

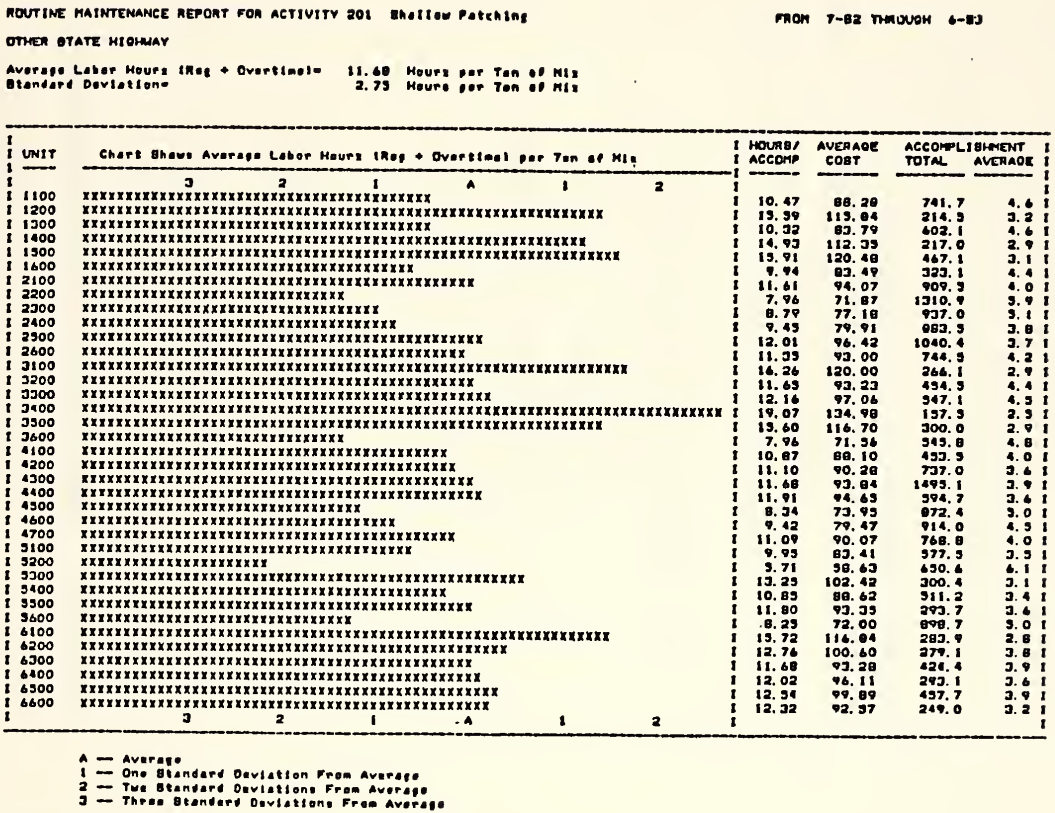
A complete description of the program including its use and outputs is given in Appendix A.

Productivity Monitoring Procedure

There are five basic steps in the productivity monitoring procedure for a given maintenance activity. The procedure is illustrated in Figure 4.5.

1. Identify deviate units. This is done by using the computer program to determine which subdistricts are deviate with respect to average unit cost. The analysis may be done for an entire year or for any number of months. One method is to divide the fiscal year into a number of periods, say six, and run the program for each two-month period as well as the entire year. This way, subdistricts that are consistently deviate throughout the year may be spotted, or seasonal trends may be revealed.

2. Analyze labor and material factors. The program calculates several factors describing the use of labor and materials by each subdistrict, and plots them in bar chart



```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                    FROM  7-82 THROUGH  6-83

OTHER STATE HIGHWAY

Average Labor Hours (Reg + Overtime)=   11.60  Hours per Ton of Mix
Standard Deviation=                      2.75  Hours per Ton of Mix

-----------------------------------------------------------------------------------------------------------------------
I                                                                                          I HOURS/  AVERAGE   ACCOMPLISHMENT  I
I UNIT    Chart Shows Average Labor Hours (Reg + Overtime) per Ton of Mix                  I ACCOMP  COST      TOTAL  AVERAGE  I
I ----    -------------------------------------------------------------------------------  I ------  -------   -----  -------  I
I                     3          2          1          A          1          2             I                                   I
I  1100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I  10.47    88.28    741.7     4.6 I
I  1200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   I  15.59   115.84    214.5     3.2 I
I  1300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I  10.32    83.79    602.1     4.6 I
I  1400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I  14.93   112.35    217.0     2.9 I
I  1500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I  15.91   120.48    467.1     3.1 I
I  1600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I   9.94    83.49    323.1     4.4 I
I  2100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I  11.61    94.07    909.5     4.0 I
I  2200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                  I   7.96    71.87   1310.9     5.9 I
I  2300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                              I   8.79    77.18    937.0     5.1 I
I  2400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                            I   9.45    79.91    883.5     3.8 I
I  2500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I  12.01    96.42   1040.4     3.7 I
I  2600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                    I  11.35    93.00    744.5     4.2 I
I  3100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I  16.26   120.00    266.1     2.9 I
I  3200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I  11.65    93.23    454.5     4.4 I
I  3300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I  12.16    97.06    547.1     4.5 I
I  3400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX     I  19.07   134.98    157.5     2.5 I
I  3500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   I  15.60   116.70    300.0     2.9 I
I  3600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                  I   7.96    71.56    545.8     4.8 I
I  4100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I  10.87    88.10    453.5     4.0 I
I  4200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                     I  11.10    90.28    737.0     3.6 I
I  4300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I  11.68    93.84   1495.1     3.9 I
I  4400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I  11.91    94.65    594.7     3.6 I
I  4500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                I   8.34    73.95    872.4     5.0 I
I  4600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                            I   9.42    79.47    914.0     4.5 I
I  4700   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                     I  11.09    90.07    768.8     4.0 I
I  5100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I   9.95    83.41    577.5     3.5 I
I  5200   XXXXXXXXXXXXXXXXXXXXXX                                                           I   5.71    58.63    650.6     6.1 I
I  5300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I  13.25   102.42    300.4     3.1 I
I  5400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I  10.85    88.62    511.2     3.4 I
I  5500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I  11.80    93.35    293.7     3.6 I
I  5600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                I   8.25    72.00    898.7     5.0 I
I  6100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   I  15.72   116.84    283.9     2.8 I
I  6200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I  12.76   100.60    279.1     3.8 I
I  6300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I  11.68    93.28    424.4     3.9 I
I  6400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I  12.02    96.11    293.1     3.6 I
I  6500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I  12.54    99.89    457.7     3.9 I
I  6600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I  12.32    92.57    249.0     3.2 I
I                     3          2          1         .A          1          2             I                                   I
-----------------------------------------------------------------------------------------------------------------------

          A -- Average
          1 -- One Standard Deviation From Average
          2 -- Two Standard Deviations From Average
          3 -- Three Standard Deviations From Average
```

Figure 4.4. Example of Bar Chart for Labor Hours

I. IDENTIFY DEVIATE UNITS

Use computer program to spot deviate units

II. ANALYZE LABOR AND MATERIAL USE

Review bar charts from program

III. REVIEW AND ANALYZE EQUIPMENT RECORDS

Manually review crew day cards for units identified in Step I

Compare equipment use with factors from Step II

IV. CONDUCT FIELD INSPECTIONS

Observe work methods of crews

Note highway conditions

Note equipment and material factors not apparent in Step II

Discuss procedures, conditions, etc. with foremen

V. DRAW CONCLUSIONS

Base conclusions on analysis in Steps II, III, and IV

If corrective action is appropriate, take action and monitor affected units

Figure 4.5. Productivity Monitoring Procedure

form. These factors include the average crew size, average number of labor hours per unit of accomplishment, average amount of a specified material used per accomplishment unit, average daily accomplishment, and total accomplishment during the analysis period. By examining these charts, along with the chart of average cost per accomplishment unit, some apparent relations between the factors and cost may be found, providing insights to the reasons for the high and low costs. For this study, a statistical analysis of these data was conducted to determine if trends spotted by visual examination of the charts corresponded with those indicated by statistical analysis.

3. Review equipment records. A manual review of a sample of crew day cards must be made to determine the type and amount of equipment used. Equipment information can help determine if procedures outlined in the performance standards are being followed and may provide clues to the quality of work. For example, a shallow patching crew with only a pickup truck would be expected to perform lower quality work than a crew that uses a portable patcher and roller. Examination of these records may indicate an equipment availability or scheduling problem.

4. Conduct field inspections. Because there is no direct indication of quality or roadway conditions available from the crew day card records, inspection of crews at work in the subdistricts under study is needed. A subdistrict

with a high average unit cost may be performing higher quality work than a district with a low unit cost. Or, using shallow patching as an example, the roadways in a low-cost subdistrict may be in worse condition than those in a high-cost subdistrict. The unit of measure for shallow patching is tons of bituminous mix placed, and labor is the highest percentage of the cost. Consider a crew in a subdistrict whose roads are in fairly good condition. The crew is assigned to perform shallow patching for the day, so the workers assemble their equipment, and patrol their assigned highway looking for holes to patch. The highway has relatively small holes that are spaced several hundred yards apart. The crew needs only a few minutes to fill each hole, but spends a large part of its time travelling between holes. It ends up producing a relatively small number of tons of patching. Now consider a second crew of the same size in a subdistrict whose roads are in worse condition. On its patrol, this crew finds relatively large holes that are spaced fairly close together. This crew would spend less time travelling, and more time actually filling holes. Thus, the second crew would have a higher accomplishment and lower average cost, simply because its highway was in worse condition than that of the first crew.

One may speculate about these possibilities based on information provided by the crew day card data, but field visits are needed for confirmation.

Discussion with subdistrict personnel may reveal special circumstances, or innovative techniques which may have a potential for application in other subdistricts.

5. Draw conclusions and take appropriate action. Based upon findings in the steps above, it can be concluded if in fact the deviate cost experienced by the units under study is the result of problems which should be corrected, a result of innovative techniques which should be shared with other subdistricts, or a result of special circumstances. If some action is deemed appropriate, it should be taken, along with continued monitoring of the subdistricts affected to evaluate the effect of the action.

Demonstration of Procedure with Crack Sealing

An analysis for activity 207, crack sealing, in fiscal year 1982-83 was conducted to demonstrate the procedure.

Identification of Deviate Units

Deviate subdistricts were identified by running the computer program for six consecutive two-month periods and for the entire fiscal year. Results of this analysis are summarized in Figure 4.6. Seven subdistricts fell into the high-cost group, while eight fell into the low-cost group.

District	Crawfordsville						Fort Wayne						Greenfield						LaPorte							Seymour						Vincennes					
Subdistrict	11	12	13	14	15	16	21	22	23	24	25	26	31	32	33	34	35	36	41	42	43	44	45	46	47	51	52	53	54	55	56	61	62	63	64	65	66
Period																																					
July-Aug. '82																																					
Sept.-Oct. '82				+	+	+			-											+														-			
Nov.-Dec. '82				+		+	+											+		+								-		-	-			-		-	-
Jan.-Feb. '83			+							+		-			+	+		+	+			-		+				-		-							-
Mar.-Apr. '83			+	-						+	+															+										-	
May-June '83																						+		+													
July '82-June '83					+		+		-		+							+	+	+		+						-		-	-		-	-		-	-

- Indicates that Productivity (cost in $/lane-mile) was below the average for all subdistricts by at least one standard deviation.

+ Indicates that Productivity (cost in $/lane-mile) was above the average for all subdistricts by at least one standard deviation.

Period	Average Cost $/lane-mile	Standard Deviation $/lane-mile
July-Aug. '82	No work this period	-----
Sept.-Oct. '82	303.03	140.65
Nov.-Dec. '82	269.84	79.87
Jan.-Feb. '83	283.03	94.32
Mar.-Apr. '83	259.33	108.58
May-June '83	300.49	134.25
July '82-June '83	262.09	56.37

Figure 4.6. Deviation Analysis Summary for FY 1982-83
Activity 207 -- Crack Sealing (Total Highway System)

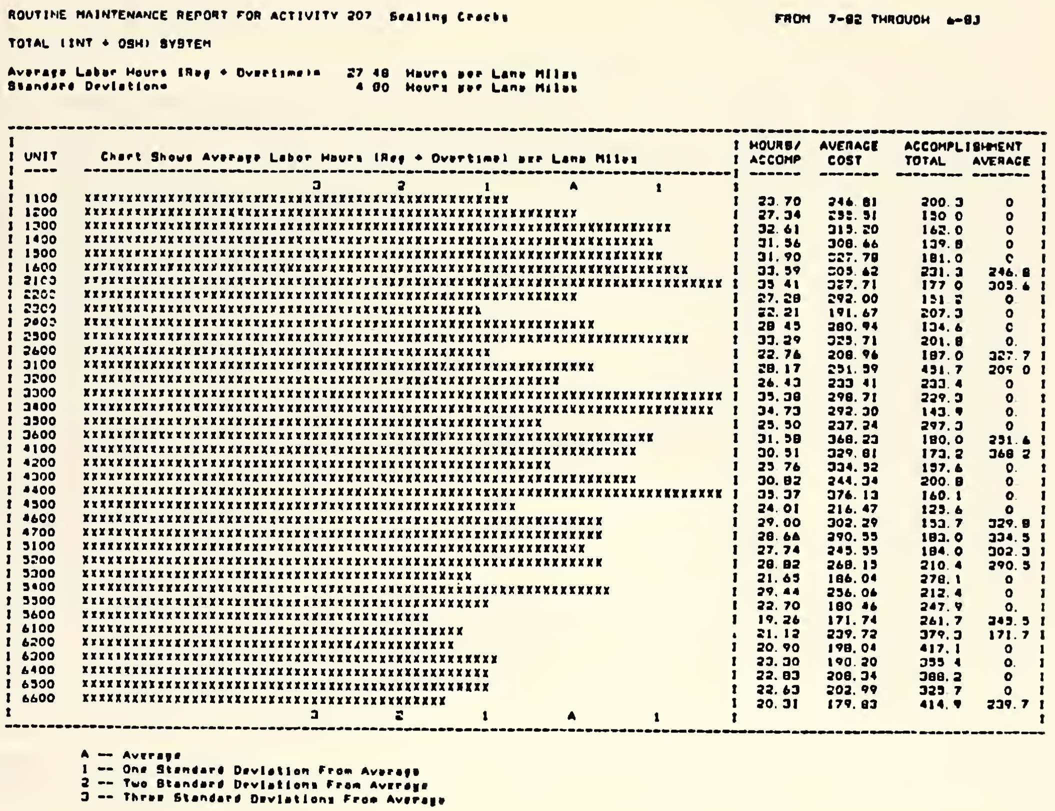
Analysis of Labor and Material Factors

Figures 4.7 through 4.12 are the bar charts showing the average cost and average labor hours per ton of mix, total accomplishment, average daily accomplishment, average crew size, and average quantity of material 4431 (bituminous material) per lane mile for the entire fiscal year. Examination of these charts indicates that the high-cost subdistricts tend to have above average labor hours per lane-mile, and low-cost subdistricts tend to have below average labor hours per lane-mile. In general, low-cost subdistricts tend to do more total crack sealing than subdistricts in the high-cost group, indicating there may be an economy of scale. As expected, there appears to be a strong relation between cost and average accomplishment, with the low-cost subdistricts exhibiting a higher average accomplishment than the high-cost subdistricts. The bar charts do not clearly indicate a relation between crew size and cost, but there seems to be a strong relation between the amount of bituminous material used per lane-mile and cost, with the low-cost subdistricts using less material than the high-cost subdistricts.

A statistical analysis was conducted to confirm these trends using data from the six two-month period analyses. An analysis of variance with covariates was used. For each month/subdistrict combination, there was one set of observations of cost and the various factors. The 37

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 207  Sealing Cracks                             FROM  7-82 THROUGH  6-83

TOTAL (INT + OSH) SYSTEM

Average Cost=        262.09  Dollars per Lane Miles
Standard Deviation=   36.37  Dollars per Lane Miles

--------------------------------------------------------------------------------------------------------------------------
I                                                                                    I AVERAGE   CREW    ACCOMPLISHMENT  I
I UNIT         Chart Shows Average Cost in Dollars per Lane Miles                    I  COST     DAYS    TOTAL   AVERAGE I
I ----   ----------------------------------------------------------------------------I -------  ------  -------- ------- I
I                          3         2          1          A           1          2  I                                   I
I 1100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I  246.81      72     200.3     2.8 I
I 1200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I  255.51      53     150.0     2.8 I
I 1300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX             I  315.20      78     162.0     2.1 I
I 1400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX              I  308.66      60     139.8     2.3 I
I 1500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX           I  327.78      83     181.0     2.2 I
I 1600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX               I  305.62      84     231.3     2.8 I
I 2100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX           I  327.71      84     177.0     2.1 I
I 2200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I  292.00      51     151.2     3.0 I
I 2300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I  191.67      57     207.3     3.6 I
I 2400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I  200.94      47     134.6     2.9 I
I 2500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX           I  325.71      82     201.8     2.5 I
I 2600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I  208.96      59     187.0     3.2 I
I 3100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I  251.59     157     451.7     2.9 I
I 3200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I  233.41      77     233.4     3.0 I
I 3300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I  298.71      89     229.3     2.6 I
I 3400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I  292.30      65     143.9     2.2 I
I 3500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I  237.24     105     297.3     2.8 I
I 3600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   I  368.23      89     180.0     2.0 I
I 4100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX          I  329.81      62     173.2     2.8 I
I 4200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX         I  334.52      51     157.6     3.1 I
I 4300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I  244.34      73     200.8     2.8 I
I 4400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX I  376.13      70     160.1     2.3 I
I 4500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I  216.47      40     125.6     3.1 I
I 4600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I  302.29      46     133.7     3.3 I
I 4700   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I  290.95      65     183.0     2.8 I
I 5100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I  249.55      74     184.0     2.5 I
I 5200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                       I  268.15      74     210.4     2.8 I
I 5300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                       I  186.04      81     278.1     3.4 I
I 5400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I  256.06      71     212.4     3.0 I
I 5500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I  180.46      67     247.9     3.7 I
I 5600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I  171.78      66     261.7     4.0 I
I 6100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I  239.72     102     379.3     3.7 I
I 6200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                     I  198.04     120     417.1     3.5 I
I 6300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                    I  198.20     113     353.4     3.2 I
I 6400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I  208.34     111     388.2     3.5 I
I 6500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                    I  202.99      91     325.7     3.6 I
I 6600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I  179.83     124     414.9     3.3 I
I                          3         2          1          A           1          2  I                                   I
--------------------------------------------------------------------------------------------------------------------------

         A -- Average
         1 -- One Standard Deviation From Average
         2 -- Two Standard Deviations From Average
         3 -- Three Standard Deviations From Average
```

Figure 4.7. Activity 207 Average Cost Chart for FY 1982-83



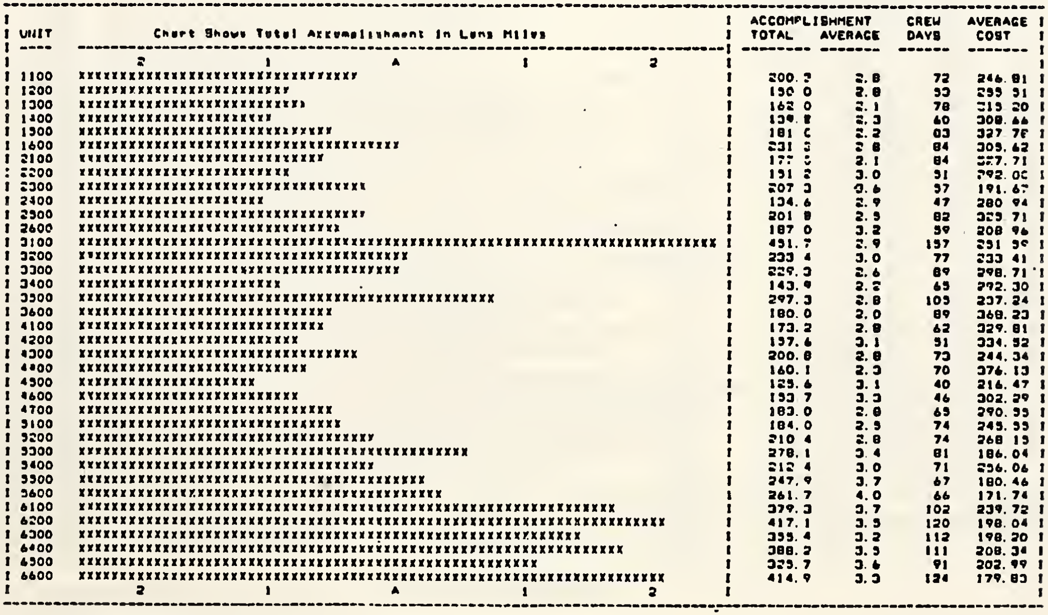
```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 207  Sealing Cracks                                  FROM  7-82 THROUGH  6-83

TOTAL (INT + OSH) SYSTEM

Average Labor Hours (Reg + Overtime)=   27.48  Hours per Lane Miles
Standard Deviation=                      4.00  Hours per Lane Miles

----------------------------------------------------------------------------------------------------------------------------
I                                                                                     I HOURS/   AVERAGE    ACCOMPLISHMENT  I
I UNIT       Chart Shows Average Labor Hours (Reg + Overtime) per Lane Miles          I ACCOMP    COST      TOTAL  AVERAGE  I
I ----    ----------------------------------------------------------------------------I ------  -------   -------  -------  I
I                                    3         2         1         A         1        I                                     I
I 1100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I  23.70   246.81     200.3        0  I
I 1200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I  27.34   232.51     150.0        0  I
I 1300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX       I  32.61   315.20     162.0        0  I
I 1400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX         I  31.56   308.66     139.8        0  I
I 1500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX        I  31.90   327.78     181.0        0  I
I 1600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX     I  33.59   305.62     231.3    246.8  I
I 2100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX I  35.41   327.71     177.0    305.6  I
I 2200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I  27.28   292.00     151.2       0.  I
I 2300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I  22.21   191.67     207.3        0  I
I 2400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I  28.45   280.94     134.6        0  I
I 2500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX     I  33.29   325.71     201.8       0.  I
I 2600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I  22.76   208.96     187.0    327.7  I
I 3100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I  28.17   251.59     451.7    209.0  I
I 3200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I  26.43   233.41     233.4        0  I
I 3300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX I  35.38   298.71     229.3       0.  I
I 3400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  I  34.73   292.30     143.9       0.  I
I 3500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I  25.50   237.24     297.3        0  I
I 3600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX         I  31.58   368.23     180.0    251.6  I
I 4100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX           I  30.51   329.81     173.2    368.2  I
I 4200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I  25.76   334.52     157.6       0.  I
I 4300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX           I  30.82   244.34     200.8       0.  I
I 4400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX I  35.37   376.13     160.1       0.  I
I 4500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I  24.01   216.47     125.6        0  I
I 4600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX               I  29.00   302.29     153.7    329.8  I
I 4700    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX               I  28.64   290.55     183.0    334.5  I
I 5100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                 I  27.74   245.55     184.0    302.3  I
I 5200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX               I  28.82   268.15     210.4    290.5  I
I 5300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I  21.65   186.04     278.1        0  I
I 5400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX              I  29.44   256.06     212.4        0  I
I 5500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I  22.70   180.46     247.9       0.  I
I 5600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I  19.26   171.74     261.7    345.5  I
I 6100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I  21.12   239.72     379.3    171.7  I
I 6200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I  20.90   198.04     417.1        0  I
I 6300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I  23.30   190.20     355.4       0.  I
I 6400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I  22.83   208.34     388.2        0  I
I 6500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I  22.63   202.99     325.7        0  I
I 6600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                         I  20.31   179.83     414.9    239.7  I
I                                    3         2         1         A         1        I                                     I
----------------------------------------------------------------------------------------------------------------------------

          A -- Average
          1 -- One Standard Deviation From Average
          2 -- Two Standard Deviations From Average
          3 -- Three Standard Deviations From Average
```

Figure 4.8. Activity 207 Average Labor Hours Chart for FY 1982-83

ROUTINE MAINTENANCE REPORT FOR ACTIVITY 207 Sealing Cracks FROM 7-82 THROUGH 6-83

TOTAL (INT + OSHI SYSTEM

Average Total Accomplishment= 229.4 Lane Miles
Standard Deviation= 89.5 Lane Miles



UNIT	ACCOMPLISHMENT TOTAL	ACCOMPLISHMENT AVERAGE	CREW DAYS	AVERAGE COST
1100	200.2	2.8	72	246.81
1200	150.0	2.8	53	255.51
1300	162.0	2.1	78	315.20
1400	139.8	2.3	60	308.66
1500	181.0	2.2	83	327.76
1600	231.3	2.8	84	305.62
2100	177.5	2.1	84	327.71
2200	151.2	3.0	51	292.00
2300	207.3	3.6	57	191.67
2400	134.6	2.9	47	280.94
2500	201.8	2.5	82	325.71
2600	187.0	3.2	59	208.96
3100	451.7	2.9	157	251.59
3200	233.4	3.0	77	233.41
3300	229.3	2.6	89	298.71
3400	143.9	2.2	65	272.30
3500	297.3	2.8	105	237.24
3600	180.0	2.0	89	368.23
4100	173.2	2.8	62	329.81
4200	157.6	3.1	51	334.52
4300	200.8	2.8	73	244.34
4400	160.1	2.3	70	376.13
4500	125.6	3.1	40	216.47
4600	153.7	3.3	46	302.29
4700	183.0	2.8	65	290.55
5100	184.0	2.5	74	245.55
5200	210.4	2.8	74	268.15
5300	278.1	3.4	81	186.04
5400	212.4	3.0	71	256.06
5500	247.9	3.7	67	180.46
5600	261.7	4.0	66	171.74
6100	379.3	3.7	102	239.72
6200	417.1	3.5	120	198.04
6300	355.4	3.2	112	198.20
6400	388.2	3.5	111	208.34
6500	325.7	3.6	91	202.99
6600	414.9	3.3	124	179.83

A -- Average
1 -- One Standard Deviation From Average
2 -- Two Standard Deviations From Average
3 -- Three Standard Deviations From Average

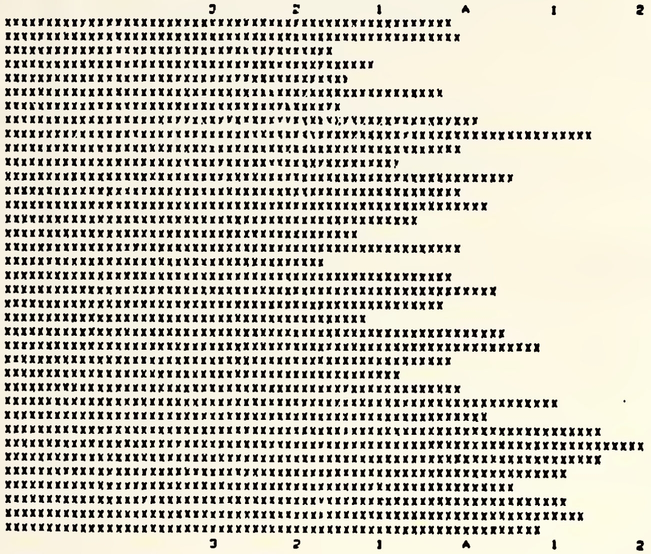
Figure 4.9. Activity 207 Total Accomplishment Chart for FY 1982-83

ROUTINE MAINTENANCE REPORT FOR ACTIVITY 207 Sealing Cracks FROM 7-82 THROUGH 6-83

TOTAL IINT + OSHI SYSTEM

Average Daily Accomplishment= 2.9 Lane Miles
Standard Deviation= 0.5 Lane Miles

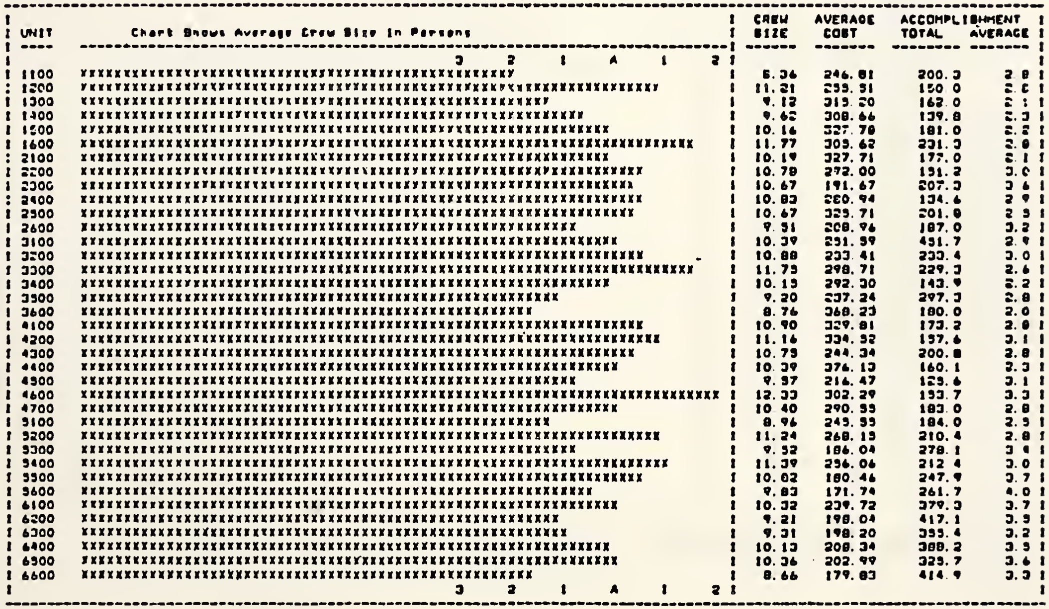
Chart Shows Average Daily Accomplishment in Lane Miles



UNIT	ACCOMPLISHMENT AVERAGE	ACCOMPLISHMENT TOTAL	CREW DAYS	AVERAGE COST
1100	2.8	200.3	72	246.81
1200	2.8	150.0	53	255.51
1300	2.1	162.0	78	313.20
1400	2.3	139.8	60	308.66
1500	2.2	181.0	83	327.78
1600	2.8	231.3	84	309.62
2100	2.1	177.0	84	327.71
2200	3.0	151.2	51	292.05
2300	3.6	207.3	57	191.67
2400	2.9	134.6	47	280.94
2500	2.5	201.8	82	325.71
2600	3.2	187.0	59	208.96
3100	2.9	451.7	157	251.59
3200	3.0	233.4	77	233.41
3300	2.6	229.3	89	298.78
3400	2.2	143.9	65	292.30
3500	2.8	297.3	105	237.24
3600	2.0	180.0	89	369.23
4100	2.8	173.2	62	329.81
4200	3.1	157.6	51	334.52
4300	2.8	200.8	73	244.34
4400	2.3	160.1	70	376.13
4500	3.1	125.6	40	216.47
4600	3.3	153.7	46	302.29
4700	2.8	183.0	65	290.55
5100	2.5	184.0	74	245.55
5200	2.8	210.4	74	268.15
5300	3.4	278.1	81	186.04
5400	3.0	212.4	71	236.06
5500	3.7	247.9	67	180.46
5600	4.0	261.7	66	171.74
6100	3.7	379.3	102	239.72
6200	3.5	417.1	120	198.04
6300	3.2	355.4	112	198.20
6400	3.5	388.2	111	208.34
6500	3.6	325.7	91	202.99
6600	3.3	414.9	124	179.83

A -- Average
1 -- One Standard Deviation From Average
2 -- Two Standard Deviations From Average
3 -- Three Standard Deviations From Average

Figure 4.10. Activity 207 Average Accomplishment Chart for FY 1982-83



ROUTINE MAINTENANCE REPORT FOR ACTIVITY 207 Sealing Cracks FROM 7-82 THROUGH 6-83

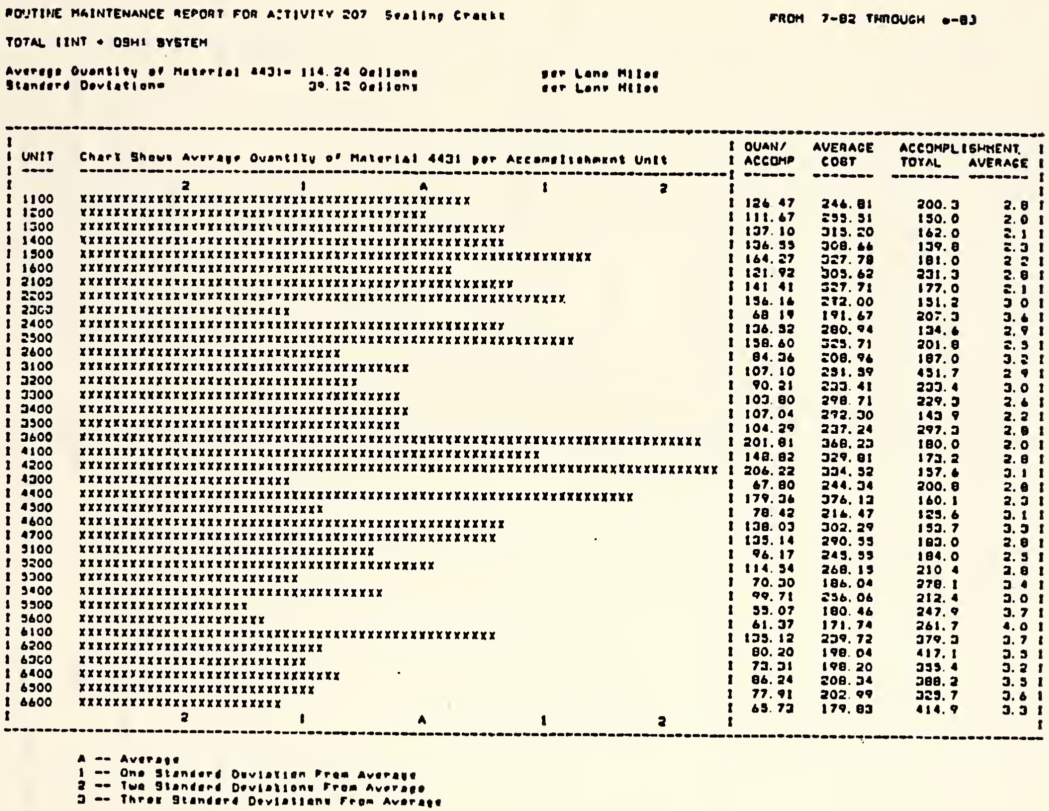
TOTAL (INT - OSH) SYSTEM

Average Crew Size= 13.25 Persons
Standard Deviation= 0.94 Persons

UNIT	CREW SIZE	AVERAGE COST	ACCOMPLISHMENT TOTAL	ACCOMPLISHMENT AVERAGE
1100	6.36	246.81	200.3	2.8
1200	11.21	255.51	150.0	2.6
1300	9.12	315.20	162.0	2.1
1400	9.62	308.66	139.8	2.3
1500	10.16	327.70	181.0	2.2
1600	11.77	305.62	231.3	2.8
2100	10.19	327.71	177.0	2.1
2200	10.78	272.00	151.2	3.0
2300	10.67	191.67	207.3	3.6
2400	10.83	260.94	134.6	2.9
2500	10.67	325.71	201.8	2.5
2600	9.51	208.96	187.0	3.2
3100	10.39	251.59	451.7	2.9
3200	10.88	233.41	233.4	3.0
3300	11.75	298.71	229.3	2.6
3400	10.15	292.30	143.9	2.2
3500	9.20	237.24	297.3	2.8
3600	8.76	368.23	180.0	2.0
4100	10.90	329.81	173.2	2.8
4200	11.16	334.52	157.6	3.1
4300	10.75	244.34	200.8	2.8
4400	10.39	376.13	160.1	3.3
4500	9.57	216.47	125.6	3.1
4600	12.33	302.29	153.7	3.3
4700	10.40	290.55	183.0	2.8
5100	8.96	245.55	184.0	2.5
5200	11.24	268.15	210.4	2.8
5300	9.52	186.04	278.1	3.4
5400	11.39	256.06	212.4	3.0
5500	10.02	180.46	247.9	3.7
5600	9.83	171.74	261.7	4.0
6100	10.32	239.72	379.3	3.7
6200	9.21	198.04	417.1	3.5
6300	9.31	198.20	355.4	3.2
6400	10.13	208.34	388.2	3.5
6500	10.36	202.99	325.7	3.6
6600	8.66	179.83	414.9	3.3

A -- Average
1 -- One Standard Deviation From Average
2 -- Two Standard Deviations From Average
3 -- Three Standard Deviations From Average

Figure 4.11. Activity 207 Average Crew Size Chart for FY 1982-83



```
ROUTINE MAINTENANCE REPORT FOR A:TIVITY 207  Sealing Cracks                                FROM  7-82 THROUGH  6-83

TOTAL (INT + OSHI SYSTEM

Average Quantity of Material 4431= 114.24 Gallons                 per Lane Miles
Standard Deviation=                 39.12 Gallons                 per Lane Miles

-------------------------------------------------------------------------------------------------------------------------
I                                                                                   I QUAN/   AVERAGE   ACCOMPLISHMENT  I
I UNIT  Chart Shows Average Quantity of Material 4431 per Accomplishment Unit       I ACCOMP   COST     TOTAL  AVERAGE  I
I ----  ----------------------------------------------------------------------------I ------  -------  -------  ------- I
I                   2             1             A             1             2       I                                   I
I 1100  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I 126.47   246.81    200.3     2.8 I
I 1200  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                    I 111.67   255.51    150.0     2.0 I
I 1300  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I 137.10   315.20    162.0     2.1 I
I 1400  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I 136.55   308.66    139.8     2.3 I
I 1500  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I 164.27   327.78    181.0     2 2 I
I 1600  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I 121.92   305.62    231.3     2.8 I
I 2100  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I 141 41   327.71    177.0     2.1 I
I 2200  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I 156.16   272.00    151.2     3 0 I
I 2300  XXXXXXXXXXXXXXXXXXXXXXXX                                                    I  68 19   191.67    207.3     3.6 I
I 2400  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I 136.32   280.94    134.6     2.9 I
I 2500  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I 158.60   325.71    201.8     2.5 I
I 2600  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                              I  84.36   208.96    187.0     3.2 I
I 3100  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I 107.10   251.59    451.7     2 9 I
I 3200  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                            I  90.21   233.41    233.4     3.0 I
I 3300  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                       I 103.80   298.71    229.3     2.6 I
I 3400  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I 107.04   272.30    143 9     2.2 I
I 3500  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                       I 104.29   237.24    297.3     2.8 I
I 3600  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    I 201.81   368.23    180.0     2.0 I
I 4100  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                       I 148.82   329.81    173.2     2.8 I
I 4200  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   I 206.22   334.52    157.6     3.1 I
I 4300  XXXXXXXXXXXXXXXXXXXXXXXX                                                    I  67.80   244.34    200.8     2.8 I
I 4400  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX            I 179.36   376.13    160.1     2.3 I
I 4500  XXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                I  78.42   216.47    125.6     3.1 I
I 4600  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I 138.03   302.29    153.7     3.3 I
I 4700  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I 135.14   290.55    183.0     2.8 I
I 5100  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I  96.17   245.55    184.0     2.5 I
I 5200  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I 114.54   268.15    210 4     2.8 I
I 5300  XXXXXXXXXXXXXXXXXXXXXXXXX                                                   I  70.30   186.04    278.1     3 4 I
I 5400  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                         I  99.71   236.06    212.4     3.0 I
I 5500  XXXXXXXXXXXXXXXXXXXX                                                        I  55.07   180.46    247.9     3.7 I
I 5600  XXXXXXXXXXXXXXXXXXXXXX                                                      I  61.37   171.74    261.7     4.0 I
I 6100  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I 135.12   239.72    379.3     3.7 I
I 6200  XXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                I  80.20   198.04    417.1     3.3 I
I 6300  XXXXXXXXXXXXXXXXXXXXXXXXXX                                                  I  73.31   198.20    335.4     3.2 I
I 6400  XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                             I  86.24   208.34    388.2     3.5 I
I 6500  XXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                I  77.91   202.99    325.7     3.6 I
I 6600  XXXXXXXXXXXXXXXXXXXXXXX                                                     I  65.73   179.83    414.9     3.3 I
I                   2             1             A             1             2       I                                   I
-------------------------------------------------------------------------------------------------------------------------

        A -- Average
        1 -- One Standard Deviation From Average
        2 -- Two Standard Deviations From Average
        3 -- Three Standard Deviations From Average
```

Figure 4.12. Activity 207 Average Material Quantity Chart for FY 1982-83

subdistricts were divided into five groups of seven each, based on the average cost for the entire year without considering variation by time period. The five groups were used as the factor levels. Data for two subdistricts were disregarded, so there would be equal cell sizes for each factor level. The dependent variable was average cost, and covariates used in the analysis were frequency (number of times activity was performed), average accomplishment, average crew size, and average amount of bituminous material used per lane-mile.

Table 4.1 presents the results of the analysis. It is desired to determine if the variables listed in the source column have an effect on the average cost. If one were to conclude that each variable in the source column does have an effect on cost, the tail probability is the probability that this conclusion is not correct. Thus, the lower the tail probability for a source, the more likely it is that the source does have an effect on average cost. If a significance level of 10 percent is selected, that is, if we accept a 10 percent risk of concluding that a variable has a significant effect when in fact it does not, the covariates with a significant effect are average accomplishment, average crew size, and average amount of bituminous material used. The beta estimate is the estimate of the standardized regression coefficient. If the beta estimate is positive, it indicates that as the value of the independent variable increases, so will the value of the dependent variable. If

Table 4.1. Analysis of Variance for Activity 207 (Crack Sealing)

Source	Degrees of Freedom	Mean Square	F	Tail Probability	Beta Estimate
Sub Group	4	1013.25	1.89	0.1168	
Covariates					
Frequency	1	832.63	1.56	0.2150	-0.16
Avg. Accomp.	1	112801.84	210.77	0.0000	-59.70
Avg. Crew	1	26799.27	50.07	0.0000	16.07
P4431	1	268228.28	501.18	0.0000	0.88
All Covariates	4	156120.58	291.71	0.0000	
Error	108	535.19			

P4431 -- Average gallons of material 4431 (Bituminous Material) used per lane-mile.

the beta estimate is negative, it indicates that as the value of the independent variable increases, the value of the dependent variable decreases.

Three of the covariates had tail probabilities of 0.00 percent: average accomplishment, average crew size, and amount of bituminous material used per lane-mile. The beta estimates indicate that as average accomplishment increases, cost decreases. As average crew size and average amount of bituminous material used per lane-mile increase, so does the average cost. The relation between crew size and cost was not apparent from the bar chart, perhaps because the variation is not very large in terms of persons. The effects of average accomplishment and amount of bituminous material used were spotted in examining the bar charts.

The analysis of variance reveals that after removing the effects due to the covariates, there is no significant variation in cost attributable to the main factor of subdistrict group. The mean square values are an indication of the relative amount of the variation in cost that is explained by the independent variables. The larger the mean square for an independent variable, the more variation in the dependent variable is explained when that independent variable is considered. The mean square values indicate that a relatively large amount of the variation in cost is removed when the amount of bituminous material used (P4431) is considered. This indicates that the subdistricts that use more material have the higher costs, a fact that may reflect the condition of the roads being sealed more than the work methods used.

Summarizing, high-cost subdistricts tend to use larger crews, seal fewer lane-miles per day, and use more bituminous material per lane-mile. This suggests that the condition of the roadway may be a main factor in governing productivity. If the roadway has a large number of cracks, it would take more time to seal them all, decreasing the number of lane-miles that a crew can seal per day, resulting in a higher unit cost.

Review of Equipment Records

A manual review of crew day card records was conducted for three subdistricts: 5300 (Columbus) from the low-cost group, 1300 (Fowler) from the average-cost group, and 4200 (Monticello) from the high-cost group. Crew day card records for the months of October, November, and December 1982 were examined to determine the type of equipment used. The performance standard for activity 207 calls for the use of pickup/pickup crew cab trucks, dump trucks, an air compressor, and a tar kettle [6].

All three subdistricts used dump trucks and pickup or pickup crew cabs 100 percent of the time. However, there were differences in the use of air compressors, tar kettles, and distributor trucks. Tables 4.2 and 4.3 summarize the

Table 4.2. Equipment Summary for Activity 207 (Crack Sealing)

Equipment	% of Time Used		
	Subdistrict Cost*		
	Low	Average	High
Dump Truck	100	100	100
Pickup/Pickup Crew Cab	100	100	100
Air Compressor	35	0	100
Tar Kettle	100	52	6
Distributor	0	48	100
Number of Observations	43	29	17

* Low -- Subdistrict 5300, Columbus
Average -- Subdistrict 1300, Fowler
High -- Subdistrict 4200, Monticello

Table 4.3. Accomplishment and Crew Size According to Equipment Use for Activity 207 (Crack Sealing)

Subdistrict 5300 (Columbus) -- Low-cost

Equipment	Average Accomplishment (Lane-miles)	Average Crew (Persons)
With Air Compressor	3.6	9.9
Without Air Compressor	3.0	8.8
Overall (Tar Kettle 100% of time)	3.2	9.2

Subdistrict 1300 (Fowler) -- Average-cost

Equipment	Average Accomplishment (Lane-miles)	Average Crew (Persons)
With Tar Kettle	2.2	9.9
With Distributor	2.5	7.7
Overall (Air Compressor 0%)	2.4	8.8

Subdistrict 4200 (Monticello) -- High-cost

Equipment	Average Accomplishment (Lane-miles)	Average Crew (Persons)
Overall (Distributor and Air Compressor 100 %)	2.3	11.1

observations. Air compressors are used to clean out cracks prior to sealing. A tar kettle contains and heats the bituminous material used for sealing. Workers fill hand-carried pots from the tar kettle and pour the bituminous material into cracks from these pots. A distributor is a tank truck which holds the bituminous material. The bituminous material is applied from a distributor via a hand-held spray bar.

The high-cost subdistrict reported use of an air compressor 100 percent of the time as contrasted with the average-cost subdistrict that never used an air compressor, and the low-cost subdistrict that reported use of an air compressor 35 percent of the time. The high-cost group reported approximately the same average accomplishment as the average-cost group, but used a larger crew. In the low-cost group, larger crew sizes and larger average accomplishment were reported when an air compressor was used than when it was not. It appears that use of an air compressor requires a larger crew, but does not seem to have an effect on average accomplishment. If this is indeed the case, use of an air compressor would cause these subdistricts to have a higher unit cost, because an extra crew member is needed to run the compressor without increasing the number of lane-miles that can be sealed in a day.

The high-cost subdistrict that used a distributor 100 percent of the time used larger crews and accomplished less per day than did the low-cost subdistrict that used a tar kettle 100 percent of the time. The average-cost subdistrict used a tar kettle 52 percent, and a distributor 48 percent of the time. When a tar kettle was used in the average-cost subdistrict, a larger crew and smaller accomplishment were reported on the average than when a distributor was used.

Based on these observations, it appears that the use of an air compressor requires a larger crew without providing an increase in daily accomplishment, leading to a higher unit cost. However, in order to properly seal cracks, they must be free of dirt and other debris. Road conditions do vary, and it may often be the case that a roadway is clean and the cracks free of debris, making it unnecessary to blow them clean. But it hardly seems likely that this would always be the case, and one would question the practice of never using an air compressor. Factors such as this can only be spotted in a manual review of records, or field inspections. No trends with respect to the use of a tar kettle versus distributor are apparent from these equipment records.

Field Inspections

One prototypical field inspection was made of a sealing crew at work in the Columbus subdistrict, selected from the low-cost group.

The observations were made on U.S. 31 north of Franklin, in Johnson County on December 13, 1983. A ten-man crew used two dump trucks, two crew cab pickups, a tar kettle, and an air compressor. A pickup pulling the air compressor was first in the process, followed by a dump truck pulling the tar kettle and three workers applying bituminous material to cracks, and three workers squeezing the material into the cracks. Next in line was a dump truck equipped with a sand spreader that backed along spreading sand on the sealed surface. Last in line was a pickup pulling an arrow board. After observing the operation and condition of the road, the unit foreman on the scene decided that it was not necessary to blow the cracks clean, allowing the compressor to be parked and the crew member operating it to help with the actual sealing.

According to the unit foreman, the bituminous material being used was unusually thick, and he expected the crew's accomplishment to be lower than normal, because the thick material is slower to pour. The bituminous material used is delivered to the subdistrict and its quality is not under their control. According to the foreman, the daily accomplishment of a crew will vary according to the conditions of the sealing material and to the severity of cracking of the roadway. The foreman indicated that U.S. 31 was one of the worst roadways in his area with respect to cracking, and need to be resurfaced. To the observer, who was more familiar with road conditions in the north-central part of the state, U.S. 31 at this location appeared to be in relatively good condition. Without an objective state-wide standard for determining pavement conditions, what constitutes a "bad" pavement as opposed to a "good" pavement is a subjective judgement. Furthermore, the severity of cracking that it takes for a road to be one of the worst in a subdistrict depends on the overall conditions of all the roads in that subdistrict. If the roads in one subdistrict exhibit overall a lower level of cracking than those in a second subdistrict, then a road that is one of the worst in the first subdistrict could be in the same condition as one of the best roads in the second. Perhaps roadways in the southern part of the state are in better condition than those in the north. Figure 4.6 indicates that seven of the eight low-cost subdistricts are in the southernmost districts, Seymour and Vincennes, while five of the seven high-cost subdistricts are in the two northernmost districts, Fort Wayne and LaPorte.

More field inspections are currently being conducted to further investigate this possibility.

Conclusions

The analysis of covariance indicates that higher costs are associated with larger crew sizes, lower average daily accomplishment, and higher rates of material use. The analysis indicates that the amount of material used per lane-mile explains a great deal of the variation in average cost. Most of the high-cost subdistricts are in the two northernmost districts, while most of the low-cost subdistricts are in the two southernmost districts. In addition, a highway in the southern half of the state that an IDOH foreman identified as one of the worst in terms of cracking appeared to be in relatively good condition to an observer who was more familiar with highways in the north-central part of the state. This suggests that there may be a difference in road conditions in the southern versus the northern part of the state, a factor on which climate could have an effect. More field inspections are being undertaken to investigate this possibility and to compare the quality of work in various subdistricts.

Demonstration of Procedure with Shallow Patching

An analysis for activity 201, shallow patching, in fiscal year 1982-83 was conducted.

Identification of Deviate Units

Deviate subdistricts were identified by running the program for two-month periods throughout the year and for the entire fiscal year. A subdistrict was defined as deviate if its average cost fell outside the range of the average plus or minus one standard deviation. The results are summarized in Figure 4.13. As seen in the figure, seven subdistricts were in the high-cost category and five in the low-cost group.

Analysis of Labor and Material Factors

Figures 4.14 through 4.19 are bar charts showing the average cost and average labor hours per ton of mix, total accomplishment, average daily accomplishment, average crew size, and average quantity of material 4431 (bituminous material) per ton of mix for the entire fiscal year. An examination of these charts indicates that the high-cost subdistricts tend to use more labor hours per ton of mix while the low-cost units use less labor. It appears that low-cost units tend to have done more total patching than high-cost units, indicating there may be some economy of scale.

As expected, there is a strong trend with respect to average daily accomplishment. Units with low daily accomplishment fall into the high-cost group, while those with high daily accomplishment tend to be found in the

District	Crawfordsville						Fort Wayne						Greenfield						LaPorte							Seymour						Vincennes					
Subdistrict	11	12	13	14	15	16	21	22	23	24	25	26	31	32	33	34	35	36	41	42	43	44	45	46	47	51	52	53	54	55	56	61	62	63	64	65	66
Period																																					
July-Aug. '82		+	-			+		-		-							+			+				-			-			+	-					+	+
Sept.-Oct. '82								-								+											-			-		+					
Nov.-Dec. '82	-	+		+	+				-														-							+			+			+	
Jan.-Feb. '83		+		+	+	-			-				+			+		-					-				-				-	+					
Mar.-Apr. '83		+		+	+	-		-					+			+	+	-									-					+					
May-June '83					+			-					+			+	+	-		-				-			-	+			-	+					
July '82-June '83		+		+	+			-	-				+			+	+	-									-				-	+					

- Indicates that Productivity (cost in $/ton) was below the average for all subdistricts by at least one standard deviation.

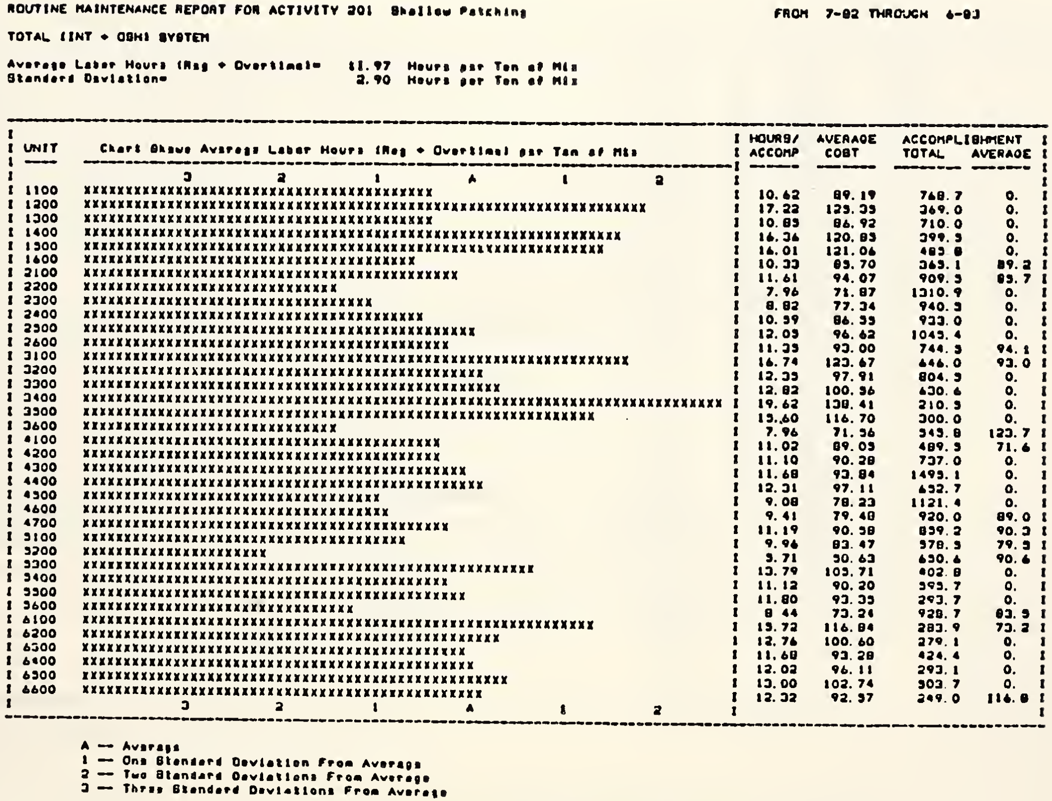
+ Indicates that Productivity (cost in $/ton) was above the average for all subdistricts by at least one standard deviation.

Period	Average Cost $/ton	Standard Deviation $/ton
July-Aug. '82	89.07	17.32
Sept.-Oct. '82	89.90	26.83
Nov.-Dec. '82	101.56	22.29
Jan.-Feb. '83	110.86	25.29
Mar.-Apr. '83	98.52	24.54
May-June '83	90.31	17.65
July '82-June '83	95.21	16.99

Figure 4.13. Deviation Analysis Summary for FY 1982-83
Activity 201 -- Shallow Patching (Total Highway System)

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                        FROM  7-82 THROUGH  6-83

TOTAL (INT + OSHI SYSTEM

Average Cost=          93.21  Dollars per Ton of Mix
Standard Deviation=    16.99  Dollars per Ton of Mix

-------------------------------------------------------------------------------------------------------------------------------
I                                                                                     I AVERAGE     CREW    ACCOMPLISHMENT   I
I UNIT          Chart Shows Average Cost In Dollars per Ton of Mix                    I  COST       DAYS      TOTAL   AVERAGE I
I ----   -----------------------------------------------------------------------------I -------   ------  ---------   ------- I
I                               3        2        1        A        1        2        I                                       I
I 1100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I   89.19      167      768.7       4.6 I
I 1200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX         I  125.35      109      369.0       3.4 I
I 1300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I   86.92      161      710.0       4.4 I
I 1400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX            I  120.85      140      399.5       2.9 I
I 1500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX           I  121.06      160      485.8       3.0 I
I 1600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I   85.70       84      365.1       4.3 I
I 2100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   94.07      228      909.5       4.0 I
I 2200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I   71.87      221     1310.9       5.9 I
I 2300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I   77.34      185      940.5       5.1 I
I 2400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I   86.55      264      933.0       3.5 I
I 2500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I   96.62      283     1045.4       3.7 I
I 2600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I   93.00      178      744.5       4.2 I
I 3100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX          I  123.67      231      646.0       2.8 I
I 3200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I   97.91      199      804.5       4.0 I
I 3300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                       I  100.56      151      630.6       4.2 I
I 3400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  I  138.41       87      210.5       2.4 I
I 3500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX              I  116.70      103      300.0       2.9 I
I 3600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I   71.56      114      545.8       4.8 I
I 4100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I   89.05      123      489.5       4.0 I
I 4200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   90.28      206      737.0       3.6 I
I 4300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   93.84      380     1495.1       3.9 I
I 4400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I   97.11      180      652.7       3.6 I
I 4500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I   78.23      235     1121.4       4.8 I
I 4600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I   79.48      206      920.0       4.5 I
I 4700   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   90.58      213      859.2       4.0 I
I 5100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I   83.47      165      578.5       3.5 I
I 5200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                             I   58.63      107      650.6       6.1 I
I 5300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I  105.71      137      402.8       2.9 I
I 5400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   90.20      172      595.7       3.5 I
I 5500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   93.35       81      293.7       3.6 I
I 5600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                     I   73.24      190      928.7       4.9 I
I 6100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX              I  116.84      100      283.9       2.8 I
I 6200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                       I  100.60       74      279.1       3.8 I
I 6300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   93.28      109      424.4       3.9 I
I 6400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I   96.11       82      293.1       3.6 I
I 6500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I  102.74      132      503.7       3.8 I
I 6600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I   92.57       79      249.0       3.2 I
I                               3        2        1        A        1        2        I                                       I
-------------------------------------------------------------------------------------------------------------------------------

         A -- Average
         1 -- One Standard Deviation From Average
         2 -- Two Standard Deviations From Average
         3 -- Three Standard Deviations From Average
```

Figure 4.14. Activity 201 Average Cost Chart for FY 1982-83



ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201 Shallow Patching FROM 7-82 THROUGH 6-83

TOTAL (INT + OSH) SYSTEM

Average Labor Hours (Reg + Overtime)= 11.97 Hours per Ton of Mix
Standard Deviation= 2.90 Hours per Ton of Mix

UNIT	HOURS/ ACCOMP	AVERAGE COST	ACCOMPLISHMENT TOTAL	ACCOMPLISHMENT AVERAGE
1100	10.62	89.19	768.7	0.
1200	17.22	125.35	369.0	0.
1300	10.85	86.92	710.0	0.
1400	16.36	120.85	399.5	0.
1500	16.01	121.06	485.8	0.
1600	10.33	85.70	365.1	89.2
2100	11.61	94.07	909.5	85.7
2200	7.96	71.87	1310.9	0.
2300	8.82	77.34	940.3	0.
2400	10.59	86.35	933.0	0.
2500	12.05	96.62	1045.4	0.
2600	11.35	93.00	744.5	94.1
3100	16.74	123.67	646.0	93.0
3200	12.35	97.91	804.5	0.
3300	12.82	100.56	630.6	0.
3400	19.62	138.41	210.5	0.
3500	15.60	116.70	300.0	0.
3600	7.96	71.56	545.8	123.7
4100	11.02	89.05	489.5	71.6
4200	11.10	90.28	737.0	0.
4300	11.68	93.84	1495.1	0.
4400	12.31	97.11	652.7	0.
4500	9.08	78.23	1121.4	0.
4600	9.41	79.48	920.0	89.0
4700	11.19	90.58	859.2	90.3
5100	9.96	83.47	578.5	79.5
5200	5.71	50.63	650.6	90.6
5300	13.79	105.71	402.8	0.
5400	11.12	90.20	595.7	0.
5500	11.80	93.35	293.7	0.
5600	8 44	73.24	928.7	83.5
6100	15.72	116.84	283.9	73.2
6200	12.76	100.60	279.1	0.
6300	11.68	93.28	424.4	0.
6400	12.02	96.11	293.1	0.
6500	13.00	102.74	503.7	0.
6600	12.32	92.57	249.0	116.8

Chart Shows Average Labor Hours (Reg + Overtime) per Ton of Mix
3
2
1
A
1
2

A — Average
1 — One Standard Deviation From Average
2 — Two Standard Deviations From Average
3 — Three Standard Deviations From Average

Figure 4.15. Activity 201 Average Labor Hours Chart for FY 1982-83

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                    FROM  7-82 THROUGH  6-83

TOTAL (INT + OSH) SYSTEM

Average Total Accomplishment=     645.3  Ton of Mix
Standard Deviation=               309.3  Ton of Mix


-----------------------------------------------------------------------------------------------------------------------------
I                                                                                          I
I UNIT              Chart Shows Total Accomplishment in Ton of Mix                         I ACCOMPLISHMENT      CREW   AVERAGE I
I ----    ---------------------------------------------------------------------------------I  TOTAL   AVERAGE    DAYS     COST   I
I         2               1               A               1               2                I --------  -------  ------  -------- I
I 1100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I   768.7      4.6     167     89.19 I
I 1200    XXXXXXXXXXXXXXXXXXX                                                              I   369.0      3.4     109    123.35 I
I 1300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                             I   710.0      4.4     161     86.92 I
I 1400    XXXXXXXXXXXXXXXXXXXX                                                             I   399.5      2.9     140    120.85 I
I 1500    XXXXXXXXXXXXXXXXXXXXXXXX                                                         I   485.8      3.0     160    121.06 I
I 1600    XXXXXXXXXXXXXXXXXX                                                               I   365.1      4.3      84     85.70 I
I 2100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I   909.5      4.0     228     94.07 I
I 2200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX               I  1310.9      5.9     221     71.87 I
I 2300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I   940.5      5.1     185     77.34 I
I 2400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I   933.0      3.5     264     86.55 I
I 2500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I  1045.4      3.7     283     96.62 I
I 2600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                            I   744.5      4.2     178     93.00 I
I 3100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                 I   646.0      2.8     231    123.67 I
I 3200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                         I   804.5      4.0     199     97.91 I
I 3300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                 I   630.6      4.2     151    100.56 I
I 3400    XXXXXXXXXXX                                                                      I   210.5      2.4      87    138.41 I
I 3500    XXXXXXXXXXXXXXX                                                                  I   300.0      2.9     103    116.70 I
I 3600    XXXXXXXXXXXXXXXXXXXXXXXXXXX                                                      I   545.8      4.8     114     71.56 I
I 4100    XXXXXXXXXXXXXXXXXXXXXXXXX                                                        I   489.5      4.0     123     89.05 I
I 4200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                            I   737.0      3.6     206     90.28 I
I 4300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    I  1495.1      3.9     380     93.84 I
I 4400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                I   652.7      3.6     180     97.11 I
I 4500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I  1121.4      4.8     235     78.23 I
I 4600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I   920.0      4.5     206     79.48 I
I 4700    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I   859.2      4.0     213     90.38 I
I 5100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                    I   578.5      3.5     165     83.47 I
I 5200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                I   650.6      6.1     107     58.63 I
I 5300    XXXXXXXXXXXXXXXXXXXX                                                             I   402.8      2.9     137    105.71 I
I 5400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                   I   595.7      3.5     172     90.20 I
I 5500    XXXXXXXXXXXXXXX                                                                  I   293.7      3.6      81     93.35 I
I 5600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I   928.7      4.9     190     73.24 I
I 6100    XXXXXXXXXXXXXX                                                                   I   283.9      2.8     100    116.84 I
I 6200    XXXXXXXXXXXXXX                                                                   I   279.1      3.8      74    100.60 I
I 6300    XXXXXXXXXXXXXXXXXXXXX                                                            I   424.4      3.9     109     93.28 I
I 6400    XXXXXXXXXXXXXXX                                                                  I   293.1      3.6      82     96.11 I
I 6500    XXXXXXXXXXXXXXXXXXXXXXXXX                                                        I   503.7      3.8     132    102.74 I
I 6600    XXXXXXXXXXXX                                                                     I   249.0      3.2      79     92.57 I
I         2               1               A               1               2                I                                   I
-----------------------------------------------------------------------------------------------------------------------------

          A -- Average
          1 -- One Standard Deviation From Average
          2 -- Two Standard Deviations From Average
          3 -- Three Standard Deviations From Average
```

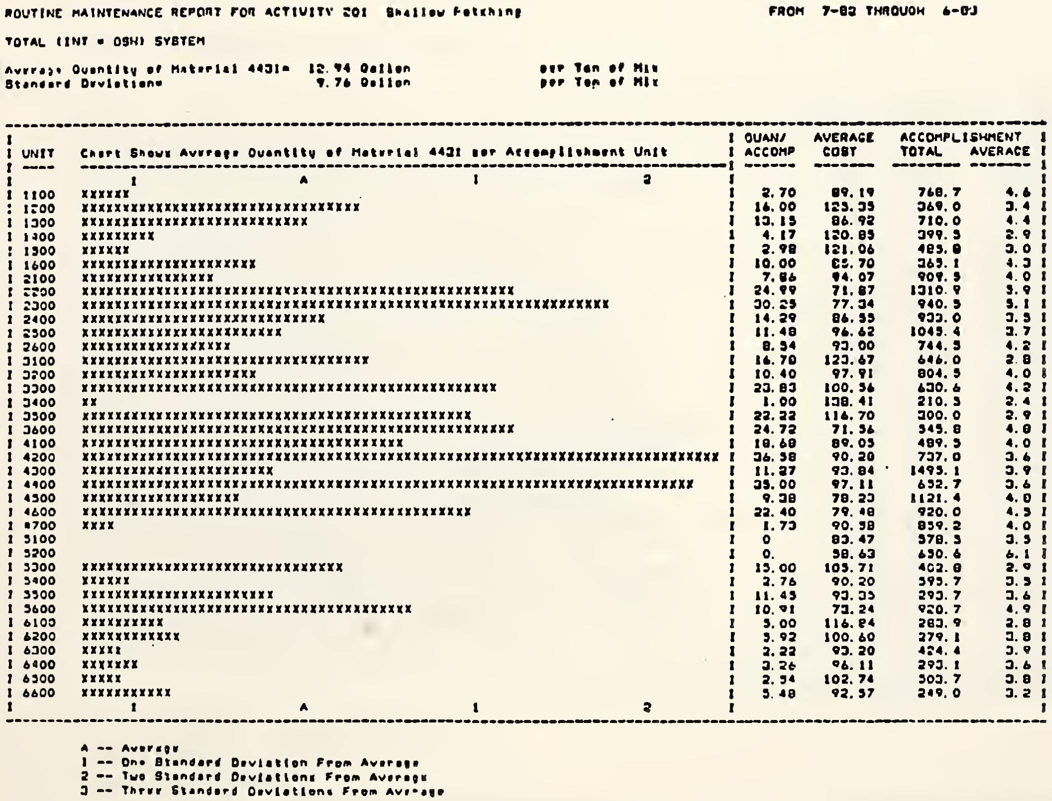
Figure 4.16. Activity 201 Total Accomplishment Chart for FY 1982-83

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                    FROM  7-82 THROUGH  6-83

TOTAL IINT - OSMI SYSTEM

Average Daily Accomplishment=       3.9  Ton of Mix
Standard Deviation=                 0.8  Ton of Mix

-------------------------------------------------------------------------------------------------------------------------
I                                                                                     I  ACCOMPLISHMENT      CREW    AVERAGE I
I UNIT       Chart Shows Average Daily Accomplishment in Ton of Mix                   I AVERAGE   TOTAL      DAYS     COST   I
I ----   ---------------------------------------------------------------------------I ------- ---------   ------  ------- I
I                         3         2         1         A         1         2          I                                     I
I 1100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   I     4.6     768.7     167    89.19 I
I 1200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I     3.4     369.0     109   125.35 I
I 1300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I     4.4     710.0     161    86.92 I
I 1400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                         I     2.9     399.5     140   120.85 I
I 1500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                       I     3.0     485.8     160   121.06 I
I 1600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I     4.3     363.1      84    85.70 I
I 2100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I     4.0     909.5     228    94 07 I
I 2200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   I     5.9    1310.9     221    71.87 I
I 2300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX             I     5.1     940.5     185    77.34 I
I 2400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I     3.5     933.0     264    86.55 I
I 2500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I     3.7    1045.4     283    96.62 I
I 2600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I     4.2     744.5     178    93.00 I
I 3100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I     2.8     646.0     231   123.67 I
I 3200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I     4.0     804.3     199    97.91 I
I 3300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I     4.2     630.6     131   100.56 I
I 3400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                              I     2.4     210.5      87   138.41 I
I 3500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I     2.9     300.0     103   116.70 I
I 3600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                 I     4.8     545.8     114    71.56 I
I 4100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I     4.0     489.5     123    89.05 I
I 4200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I     3.6     737.0     206    90.28 I
I 4300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I     3.9    1495.1     380    93.84 I
I 4400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I     3.6     652.7     180    97.11 I
I 4500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                 I     4.8    1121.4     235    78.23 I
I 4600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I     4.5     920.0     206    79.48 I
I 4700   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I     4.0     859.2     213    90.58 I
I 5100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I     3.5   . 578.5     165    83.47 I
I 5200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX I     6.1     650.6     107    98.63 I
I 5300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I     2.9     402.8     137   105.71 I
I 5400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I     3.5     595.7     172    90.20 I
I 5500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I     3.6     293.7      81    93.35 I
I 5600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I     4.9     928.7     190    73.24 I
I 6100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                         I     2.8     283.9     100   116.84 I
I 6200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I     3.8     279.1      74   100.60 I
I 6300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I     3.9     424.4     109    93.28 I
I 6400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I     3.6     293.1      82    96.11 I
I 6500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I     3.8     503.7     132   102.74 I
I 6600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                     I     3.2     249.0      79    92.57 I
I                         3         2         1         A         1         2          I                                     I
-------------------------------------------------------------------------------------------------------------------------

         A -- Average
         1 -- One Standard Deviation From Average
         2 -- Two Standard Deviations From Average
         3 -- Three Standard Deviations From Average
```

Figure 4.17. Activity 201 Average Accomplishment Chart for FY 1982-83

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                                   FROM  7-82 THROUGH  6-83

TOTAL  LINT + OSHI SYSTEM

Average Crew Size=        5.88   Persons
Standard Deviation=       0.62   Persons

------------------------------------------------------------------------------------------------------------------------------------------------
I                                                                                                      I  CREW    AVERAGE    ACCOMPLISHMENT   I
I UNIT          Chart Shows Average Crew Size in Persons                                               I  SIZE     COST      TOTAL   AVERAGE  I
I ----     --------------------------------------------------------------------------------------------I -------  -------   -------- -------  I
I                                                        3       2      1      A      1      2      3  I                                      I
I   1100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   I   6.31     89.19     768.7      4.6  I
I   1200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  I   7.81    125.35     369.0      3.4  I
I   1300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I   6.29     86.92     710.0      4.4  I
I   1400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I   6.25    120.85     399.5      2.9  I
I   1500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   I   6.39    121.06     405.8      3.0  I
I   1600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I   5.87     85.70     363.1      4.3  I
I   2100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   I   6.34     94.07     909.5      4.0  I
I   2200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I   6.29     71.87    1310.9      5.9  I
I   2300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I   6.19     77.34     940.5      5.1  I
I   2400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I   5.35     86.55     933.0      3.5  I
I   2500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I   5.83     96.62    1045.4      3.7  I
I   2600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I   6.15     93.00     744.5      4.2  I
I   3100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I   5.87    123.67     646.0      2.8  I
I   3200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   I   6.32     97.91     804.5      4.0  I
I   3300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX          I   7.09    100.56     630.6      4.2  I
I   3400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I   6.46    138.41     210.5      2.4  I
I   3500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I   5.82    116.70     300.0      2.9  I
I   3600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I   5.46     71.56     345.8      4.8  I
I   4100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   5.70     89.05     409.5      4.0  I
I   4200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I   5.44     90.20     737.0      3.6  I
I   4300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I   6.05     93.84    1495.1      3.9  I
I   4400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I   5.93     97.11     652.7      3.6  I
I   4500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   5.34     78.23    1121.4      4.8  I
I   4600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I   5.59     79.48     920.0      4.3  I
I   4700   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I   6.00     90.38     839.2      4.0  I
I   5100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I   4.52     83.47     570.5      3.5  I
I   5200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                       I   4.57     58.63     650.6      6.1  I
I   5300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I   5.31    105.71     402.0      2.9  I
I   5400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I   5.09     90.20     395.7      3.5  I
I   5500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   5.70     93.33     293.7      3.6  I
I   5600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I   5.45     73.24     920.7      4.9  I
I   6100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I   6.09    116.84     203.9      2.0  I
I   6200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I   6.03    100.60     279.1      3.0  I
I   6300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   5.74     93.28     424.4      3.9  I
I   6400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   5.49     96.11     293.1      3.6  I
I   6500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I   6.28    102.74     303.7      3.9  I
I   6600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I   5.0[illegible]     92.57     249.0      3.2  I
I                                                        3       2      1      A      1      2      3  I                                      I
------------------------------------------------------------------------------------------------------------------------------------------------

         A -- Average
         1 -- One Standard Deviation From Average
         2 -- Two Standard Deviations From Average
         3 -- Three Standard Deviations From Average
```

Figure 4.18. Activity 201 Average Crew Size Chart for FY 1982-83



```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                        FROM  7-82 THROUGH  6-83

TOTAL (INT = OSHI SYSTEM

Average Quantity of Material 4431=  12.94 Gallon          per Ton of Mix
Standard Deviation=                  9.76 Gallon          per Ton of Mix

----------------------------------------------------------------------------------------------------------------------------
I                                                                                     I QUAN/    AVERAGE    ACCOMPLISHMENT  I
I UNIT   Chart Shows Average Quantity of Material 4431 per Accomplishment Unit       I ACCOMP    COST      TOTAL   AVERAGE I
I ----   ----------------------------------------------------------------------------I ------   -------   ------- ------- I
I              1                   A                   1                   2         I                                    I
I 1100   XXXXXX                                                                      I   2.70     89.19     768.7      4.6 I
I 1200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                           I  16.00    123.35     369.0      3.4 I
I 1300   XXXXXXXXXXXXXXXXXXXXXXXXXXX                                                 I  13.15     86.92     710.0      4.4 I
I 1400   XXXXXXXXX                                                                   I   4.17    120.85     399.5      2.9 I
I 1500   XXXXXX                                                                      I   2.98    121.06     485.8      3.0 I
I 1600   XXXXXXXXXXXXXXXXXXXXX                                                       I  10.00     65.70     365.1      4.3 I
I 2100   XXXXXXXXXXXXXXXX                                                            I   7.86     94.07     909.5      4.0 I
I 2200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I  24.99     71.87    1310.9      5.9 I
I 2300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX              I  30.25     77.34     940.5      5.1 I
I 2400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                               I  14.29     86.55     933.0      3.5 I
I 2500   XXXXXXXXXXXXXXXXXXXXXXXX                                                    I  11.48     96.62    1045.4      3.7 I
I 2600   XXXXXXXXXXXXXXXXXX                                                          I   8.54     93.00     744.5      4.2 I
I 3100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I  16.70    123.67     646.0      2.8 I
I 3200   XXXXXXXXXXXXXXXXXXXXX                                                       I  10.40     97.91     804.5      4.0 I
I 3300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I  23.83    100.56     630.6      4.2 I
I 3400   XX                                                                          I   1.00    138.41     210.5      2.4 I
I 3500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I  22.22    116.70     300.0      2.9 I
I 3600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I  24.72     71.56     545.8      4.8 I
I 4100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I  18.68     89.05     489.5      4.0 I
I 4200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX I  36.58     90.20     737.0      3.6 I
I 4300   XXXXXXXXXXXXXXXXXXXXXXX                                                     I  11.27     93.84    1493.1      3.9 I
I 4400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX    I  35.00     97.11     632.7      3.6 I
I 4500   XXXXXXXXXXXXXXXXXXX                                                         I   9.38     78.23    1121.4      4.0 I
I 4600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I  22.40     79.48     920.0      4.5 I
I 4700   XXXX                                                                        I   1.73     90.58     859.2      4.0 I
I 5100                                                                               I      0     83.47     578.5      3.5 I
I 5200                                                                               I     0.     58.63     650.6      6.1 I
I 5300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                             I  15.00    105.71     402.8      2.9 I
I 5400   XXXXXX                                                                      I   2.76     90.20     595.7      3.5 I
I 5500   XXXXXXXXXXXXXXXXXXXXXXX                                                     I  11.45     93.35     293.7      3.6 I
I 5600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                     I  18.91     73.24     920.7      4.9 I
I 6100   XXXXXXXXXX                                                                  I   5.00    116.84     283.9      2.8 I
I 6200   XXXXXXXXXXXX                                                                I   5.92    100.60     279.1      3.8 I
I 6300   XXXXX                                                                       I   2.22     93.20     424.4      3.9 I
I 6400   XXXXXXX                                                                     I   3.26     96.11     293.1      3.6 I
I 6500   XXXXX                                                                       I   2.54    102.74     503.7      3.8 I
I 6600   XXXXXXXXXXX                                                                 I   5.48     92.57     249.0      3.2 I
I              1                   A                   1                   2         I                                    I
----------------------------------------------------------------------------------------------------------------------------

         A -- Average
         1 -- One Standard Deviation From Average
         2 -- Two Standard Deviations From Average
         3 -- Three Standard Deviations From Average
```

Figure 4.19. Activity 201 Average Material Quantity Chart for FY 1982-83

low-cost group. Average crew size does not vary widely in terms of numbers of persons, and no trend with respect to cost is apparent. It appears that the average amount of material 4431 (bituminous material used as a tack coat) used per ton of patching does not have an effect on the average cost.

Summarizing, a comparison of the bar charts suggests that subdistricts with low daily production and correspondingly high labor use per ton of patching fall into the high-cost group, while those with high daily production and low labor use per ton fall into the low-cost group. A statistical analysis was conducted to confirm these trends.

Data used for the analysis were from the six two-month analyses. For each two-month period and each subdistrict, there was one observation of cost, average labor hours, average accomplishment, average crew size, etc. An analysis of variance with covariates was conducted. Based on the average cost figures for the entire year, the 37 subdistricts were divided into five groups of seven each. The five groups were used as the factor levels. Data from two subdistricts were disregarded so there would be equal cell sizes for each factor level. Average cost was taken as the independent variable, and the continuous variables of frequency (number of times the activity was performed), average accomplishment, average crew size, fraction of the time hot bituminous mixture was used, and fraction of the time bituminous material was used, were the covariates. Average labor hours was not used, because this figure was used to calculate the cost directly. Furthermore, the analysis did not consider the variation by 2-month time period.

Table 4.4 presents the results of the analysis. If a significance level of 10 percent is selected, that is, if we accept a 10 percent risk of concluding that a variable has a significant effect when in fact it does not, the covariates with a significant effect are average accomplishment and average crew size. The beta estimates indicate that average cost increases as average accomplishment decreases and crew size increases.

Thus, the analysis of covariance confirms the trends spotted through examination of the bar charts. The subdistricts with higher costs have a lower daily accomplishment than those with lower costs. While there is not a large variation in crew size, larger crews are associated with higher costs. This is expected, since labor is the highest part of the cost of shallow patching.

Review of Equipment Records

A manual review of crew day card records was conducted for three subdistricts: 5200 (Bloomington) from the low-cost group, 1300 (Fowler) from the average-cost group, and 1200 (Crawfordsville) from the high-cost group. Crew day card records for the months of October, November, and December 1982 were examined to determine the type of equipment used.

Table 4.4. Analysis of Variance for Activity 201 (Shallow Patching)

Source	Degrees of Freedom	Mean Square	F	Tail Probability	Beta Estimate
Sub Group	4	923.30	7.30	0.0000	
Covariates					
Frequency	1	272.23	2.15	0.1438	-0.08
Avg. Accomp.	1	29267.40	231.55	0.0000	-12.63
Avg. Crew	1	8675.78	68.64	0.0000	10.39
F4441	1	33.65	0.27	0.6064	-1.29
F4431	1	18.74	0.15	0.7006	1.07
All Covariates	5	7370.33	58.31	0.0000	
Error	200	126.39			

F4441 -- Fraction of the time material 4441 (Hot Bituminous Mix) was used.

F4431 -- Fraction of the time material 4431 (Bituminous Material) was used

Table 4.5. Equipment Summary for Activity 201 (Shallow Patching)

Equipment	% of Time Used		
	Subdistrict Cost*		
	Low	Average	High
Dump Truck	77	82	47
Pickup/Pickup Crew Cab	100	100	100
Bituminous Mix Trailer	45	54	0
Portable Patcher	0	0	26
Number of Observations	31	39	19

* Low -- Subdistrict 5200, Bloomington
Average -- Subdistrict 1300, Fowler
High -- Subdistrict 1200, Crawfordsville

Table 4.6. Accomplishment and Crew Size According to Equipment Use for Activity 201 (Shallow Patching)

Subdistrict 5200 (Bloomington) -- Low-cost		
Equipment	Average Accomplishment (Tons)	Average Crew (Persons)
With Bit Mix Trailer	5.6	5.2
Without Bit Mix Trailer	7.1	4.5
Overall	6.4	4.8

Subdistrict 1300 (Fowler) -- Average-cost		
Equipment	Average Accomplishment (Tons)	Average Crew (Persons)
With Bit Mix Trailer	3.8	6.7
Without Bit Mix Trailer	3.8	6.4
Overall	3.8	6.6

Subdistrict 1200 (Crawfordsville) -- High-cost		
Equipment	Average Accomplishment (Tons)	Average Crew (Persons)
With Portable Patcher	3.6	9.2
Without Portable Patcher	2.0	6.0
Overall	2.4	6.8

Tables 4.5 and 4.6 summarize the observations. The performance standard for activity 201 calls for the use of a dump truck, pickup crew cab, bituminous mix trailer, and a compactor (hand or mechanical) [6].

Dump trucks were used for shallow patching most of the time in the low-cost and average-cost subdistricts (77 and 82 percent of the time, respectively) but only 42 percent of the time in the high-cost subdistrict. Pickup and/or pickup crew cab trucks were always used. The use of a mechanical compactor was only reported once, that being in the average-cost subdistrict. The largest variation was in the use of bituminous mix trailers and portable patchers.

A bituminous mix trailer consists basically of a box for storage of the mix, and propane heaters to keep the mix warm. This slows the cooling of hot mix, and helps make cold mix easier to work with in cold weather. A portable patcher consists of a rotating cylinder type heater. Cold mix is shoveled into one end and is heated to approximately 275° F by the time it exits the rotating cylinder. The heated mix is easier to work with and makes a higher quality patch than cold mix that is unheated, or just warmed in a bituminous mix trailer.

In the high-cost subdistrict, a portable patcher was used 26 percent of the time. Larger crews were used along with the patcher. The average crew size when a portable patcher was reported was 9.2 persons, compared to 6.0 when one was not used. However, average accomplishment increased when a portable patcher was used, being 3.6 tons compared with 2.2 tons otherwise. A portable patcher was reported only on Interstate repairs. This may indicate that a higher quality of repair is made on an Interstate than on an Other State Highway.

The high-cost subdistrict did not report the use of a bituminous mix trailer during these months. However, the low-cost and average-cost subdistricts did use a bituminous mix trailer, but did not use a portable patcher.

In the low-cost subdistrict, a bituminous mix trailer was used 45 percent of the time. Average crew size with a bituminous mix trailer was higher, 5.2 versus 4.5 men, and average accomplishment was lower, 5.6 versus 7.8 tons.

In the average-cost subdistrict, a bituminous mix trailer was used 54 percent of the time. However, the average crew size was only slightly higher when a trailer was used, 6.7 versus 6.4 men. Also, the average accomplishment was only slightly lower when a trailer was used, 3.7 versus 3.9 tons.

Based on the observations in these three subdistricts, it appears that the use of a portable patcher coincides with a larger crew size and higher average production within the one subdistrict studied. No such conclusion can be drawn with regard to the bituminous mix trailer. Both the low-cost and average-cost subdistricts reported using a trailer approximately half of the time. The average crew size and accomplishment in the average-cost subdistrict was approximately equal regardless of the type of equipment used, but in the low-cost subdistricts, larger crews and lower accomplishment accompanied use of a trailer.

Conclusions

It was not possible to make field inspections of shallow patching during the initial phase of the study; however, field inspections are currently being conducted in the final phase of the study. It would be premature to draw conclusions until the field studies are completed, but some preliminary observations can be made. Subdistricts that use larger than average crews and achieve lower than average daily accomplishment exhibit higher than average unit costs. When a portable patcher is used instead of a bituminous mix trailer or just a dump or pickup truck, unit costs tend to increase. No observations with respect to quality of work can be made without field inspections.

CHAPTER 5

SUMMARY AND RECOMMENDATIONS

Summary

The use of the monitoring procedure developed in this study can assist maintenance managers in improving the productivity of their operations. Using information currently collected by the IDOH Maintenance Management Information System (MMIS), a computer program developed in the study identifies subdistricts with unusually high or low resource use. Further study of these subdistricts should reveal the reasons for their deviations in resource use. If the reason for a subdistrict′s being deviate is improper work methods, corrective action can be taken. If on the other hand, a subdistrict is deviate because its managers use more efficient methods to provide the required quality of work, information about these methods may be disseminated throughout the state. Another reason for a subdistrict′s being deviate could be the physical condition of the roadway. For example, a subdistrict may have a high cost for crack sealing because its roads simply have more cracks to seal per lane-mile. No corrective action would be indicated in such a case, but these factors can be identified and considered when programming maintenance for that subdistrict.

Because the computer program was developed to use information currently available from crew day cards as entered on computer tape, there are no reports of equipment use, and no consideration of work quality in the computer reports. This makes it necessary to manually review crew day cards to gather equipment information, and to make field inspections of crews at work to evaluate the quality of work in a subdistrict. Manual inspection of crew day cards is a time-consuming procedure. With the total number of cards, it would not be possible to gather complete information. Only a sampling of cards from the few subdistricts identified for individual study would be possible. If the equipment information were included on computer tape along with the labor and material information, it could be analyzed easily by computer.

Recommendations

It is recommended that the procedure developed in this study be demonstrated further, with particular emphasis on evaluating the quality of maintenance operations. An analysis should be conducted to determine if the quality of maintenance work is correlated with the average cost of production, or other productivity measures available from the computer compilation of crew day card data.

The IDOH equipment use information that is currently gathered via crew day cards should be recorded on computer tape along with the labor and material information. This would allow equipment use to be evaluated along with labor and material information via computer, eliminating the need for manual review of crew day cards.

Because the current IDOH Maintenance Management Information System has no measure of roadway conditions or quality of maintenance, it is recommended that a systematic survey of quality or roadway conditions be considered for development and implementation. This would supply a key piece of information for productivity analysis, providing a broader information base than the limited spot inspections of crews at work. It would allow correlation of quality or roadway condition factors with the productivity measures for the entire state, instead of just for the few subdistricts singled out for individual study. This would not eliminate the need for subdistrict visits as part of the procedure, however. Observations of crews at work and discussions with personnel in subdistricts identified as deviate in resource use would still be a valuable source of insight to the reasons for the observed deviations.

A comprehensive survey of road conditions has potential use in the programming process as well. If the survey were designed to identify roadway deficiencies to which maintenance crews should respond, it could be used for determining maintenance needs, as well as an overall measure of the quality of maintenance in a subdistrict.

REFERENCES

1. Byrd, Tallamy, and MacDonald, "A Study of Highway Maintenance Quality Levels in Ohio", Falls Church, Va. December, 1970.

2. Epps, J.A., "Developing Decision-Making Maintenance Management Tools", Public Works, Vol. 107, No. 2, February, 1976.

3. Helland, H.C., "Performance Budgeting for Maintenance: A Budget Based on Quality, Quantity, and Productivity Standards", Highway Research Record 298, 1969.

4. Highway Research Board, "Interstate Highway Maintenance Requirements and Unit Maintenance Expenditure Index", NCHRP Report 42, 1967.

5. Indiana Department of .Highways, Division of Maintenance, "Cost Calculation Report for ACMMS103", 1981-82.

6. Indiana Department of Highways, Division of Maintenance, "Field Operations Handbook for Foremen", 1982-1983.

7. Indiana Department of Highways, Division of Maintenance, "Maintenance Management System Activity/Inventory Correlation", June, 1982.

8. Indiana Department of Highways, Division of Maintenance, "Maintenance Management System Maintenance Feature Inventory", April, 1982.

9. Keefe, R.A., "Proposed Framework for a Maintenance Management System", Public Works, Vol. 107, No. 1, January, 1976.

10. Leslie, A., Gibson, A., Conliffe, A.P., "The Ontario Approach to Maintenance Management", Highway Research Board Special Report 100, 1968.

11. Miller, E.L., "A Method of Measuring the Quality of Highway Maintenance", Transportation Research Record 506, 1974.

12. Poister, Theodore H., Huebner, Richard S., Gittings, Gary L., Phillips, Kathy, "Development of Performance Indicators for the Pennsylvania Department of Transportation", The Pennsylvania Transportation Institute, The Pennsylvania State University, September, 1980.

13. Sharaf, Essam A., Sinha, Kumares C., Whitmire, Clay, Yoder, Eldon J., "Field Investigation of Resource Requirements for State Highway Routine Maintenance Activities", Joint Highway Research Project; Purdue University, July, 1982.

14. Slaybaugh, N.L., Poister, T.H., Kilareski, W.P., "An Analysis of the Relationship Between TOS Condition Data and Highway Maintenance Needs", The Pennsylvania Transportation Institute, The Pennsylvania State University, March, 1983.

APPENDIX A

RMPS

Routine Maintenance Productivity Summary

USER'S MANUAL

Introduction

Program RMPS (Routine Maintenance Productivity Summary) analyzes crew day card data from the IDOH Maintenance Management System, and produces relatively simple and straight-forward reports showing various factors by which subdistrict performance may be assessed. It is assumed that the user is familiar with the IDOH Maintenance Management System.

RMPS analyzes data for one activity and time period per run. On the basis of the crew day card records, the program calculates the following factors for each subdistrict: the number of times a given activity was performed, the total amount of work accomplished in the time period under study, the average accomplishment per crew day, the average crew size, the average number of manhours (both regular and overtime) per crew day, and the number of manhours per accomplishment unit. Also determined are the percent of the time a given material is used, the average quantity of the material when it is used, and the average quantity of material per accomplishment unit, when that material is used. The average cost per accomplishment unit is calculated, along with the labor cost and material cost per accomplishment unit. A summary of accomplishment amounts, labor and material use, and cost, for each of the six districts and the state as a whole are also calculated. This information is presented in tabular form. The subdistrict summary information can also be presented in bar chart form.

After calculating the various factors for each subdistrict, RMPS takes the average cost figures for each subdistrict and calculates the average and standard deviation. Then, the average cost for each subdistrict is checked to see if it falls outside the range of the overall average plus or minus a given number of standard deviations. These deviate units are then listed.

With the results from RMPS, maintenance managers can spot subdistricts with unusually high or low resource use, total accomplishment, or average accomplishment. The various factors calculated and presented in bar chart form provide a starting point for the manager to determine reasons for the observed deviations.

Manual Organization

The remainder of this manual is organized into three main sections. First, the data base used by the program is explained. The second section explains the input preparation and program output. Output from a sample run of RMPS is included in this section, followed by an explanation of the program error messages. The third section explains the calculation procedure and program structure, and includes notes on computer resource requirements.

Data Base

The data base used by RMPS is IDOH Maintenance Division crew day card data as recorded on magnetic tape. Crew day card information may be entered as one or two "records" on the tape. A record may be thought of as one 80-character line of data, similar to a computer card with 80 columns. All maintenance activities require the utilization of labor, but all do not require the use of materials. Labor and material information are entered on two separate records. Normally, the first record entered is the labor record. If the activity also required the use of material, the material information is listed on the next record. The data entered on a labor record, their column locations on the record, and their formats are listed in Table A.1. The data entered on a material record, their column locations and formats are listed in Table A.2.

There are two other types of material and labor records. A "91" record is a labor record entered by central office personnel to correct mistakes found in the labor information entered by subdistricts. A "9m" is a similar record for materials. Because there are a relatively small number of these types of cards, and RMPS checks for gross typographical errors in entering data, the "91" and "9m"

Table A.1. Format of Labor Record

Column	Information	Format
1-2	Card type (91)	A2
3-5	Activity Number	I3
8-10	Road Class Abbr.(INT or OSH)	A3
11-14	Management Unit Identification	I4
18-23	Date(MMDDYY)	3I2
24-25	Route Type(US, SR, or I)	A2
26-28	Route Number	I3
29-30	County	I2
31-32	Crew Size	I2
33-36	Regular Hours	I4
37-40	Overtime Hours	I4
41-46	Accomplishment	F6.1

Table A.2. Format of Material Record

Column	Information	Format
1-2	Card Type (93)	A2
3-5	Activity Number	I3
8-10	Road Class Abbr.(INT or OSH)	A3
11-14	Management Unit Identification	I4
18-23	Date(MMDDYY)	3I2
24-25	Route Type(US, SR, or I)	A2
26-28	Route Number	I3
29-30	County	I2
31-34	Material Number, 1	I4
35-40	Material Quantity, 1	F6.1
41-44	Material Number, 2	I4
45-50	Material Quantity, 2	F6.1
51-54	Material Number, 3	I4
55-60	Material Quantity, 3	F6.1
61-64	Material Number, 4	I4
65-70	Material Quantity, 4	F6.1
71-74	Material Number, 5	I4
75-80	Material Quantity, 5	F6.1

records are not included in the analysis.

The data file for an entire year contains approximately 400,000 individual records. They are entered roughly in chronological order, but there is no order at all by subdistrict or activity type. Since RMPS analyzes data for only one activity during a run, it is strongly recommended that the crew day card file first be edited, and all records for the activity to be analyzed be placed in a different file. This will save a great deal of computer time during the execution of RMPS. This is particularly true if more than one run of RMPS will be made for the same activity. Thus, it is better to sort the crew day card records once, rather than requiring each run of RMPS to look through the entire data tape.

Program Input and Output

Two data files are input for RMPS. One contains the parameters that specify what calculations RMPS will make, and the other is the crew day card data file. The crew day card data file was described under "Data Base" earlier in this manual. This section describes the preparation of the input parameter file and the resulting RMPS output. The program error and data check messages are also explained.

Input Parameters

Parameters are input in the form of a data file that is read by the program. Data entry is free format. At least one space must separate parameters on the same line, and each line must contain all the parameters specified for that line as indicated below. Figure A.1 is an example parameter input file.

LINE 1 Enter the value of variable IB. IB is the first line in the crew day card file that will be examined by RMPS. If an analysis is being conducted, and the user knows the approximate location on the data file of the pertinent crew day card records, IB may be used to cause the program to skip over the first (IB - 1) records before examining the information on each line. This will decrease the program's execution time. For example, if the user knows that the data pertinent to the analysis are somewhere after the first 8000 lines, IB can be set to 8000. The program would skip the first 7999 lines without examining the information contained therein. Line 8000 would be the first one checked to see if it is pertinent to the analysis. If the location of the data is not known, IB should be set to 1. One numerical entry is made on this line.

LINE 2 Enter IDOH Maintenance Management code for the activity to be analyzed (variable IACT). Enter the number of materials that are likely to be used for this activity, the number of "specified materials" (variable NMAT). Two numerical entries are made on this line.

LINE 3 Enter the description of the activity being analyzed (variable DACT). The description must be contained in the first 40 spaces of the line and is the only entry on this line.

LINE 4 Enter the unit of measure of the activity (variable UACT). The unit name must be contained in the first 20 spaces of the line, and it is the only entry on this line.

LINE 5 Enter the IDOH Maintenance Management code of the first specified material (variable SM(1)). Enter the unit cost for the material (variable CSM(1)). Enter the maximum expected quantity of the material (variable QMAX(1)). QMAX(1) is a value that is used for checking the crew day card data. If a quantity of specified material greater than QMAX(1) is read, it is suspected that this is a typographical error, a data check message is printed, and the value is not used in the analysis. Three numerical entries are made on this line.

LINE 6 Enter the description of specified material 1 (variable DSM(1)). The description must be contained within the first 40 spaces of the line and is the only entry on this line.

LINE 7 Enter the unit of measure of specified material(1) (variable USM(1)). The description of the unit must be contained in the first 20 spaces of the line and is the only entry on this line.

The above three lines are repeated for each material to be specified. The number of materials to be specified is equal to NMAT; thus, the total number of lines used to describe the materials associated with an activity will be (3 * NMAT). After all the materials have been specified, the next line number in the file will be (LN + 1), where LN=4 + (3 * NMAT).

LINE LN+1 Enter the average regular-time wage rate per worker in $ per hour (variable CRH). Enter the average overtime wage rate per worker in $ per hour (variable COT). A total of two numerical entries are made on this line.

LINE LN+2 Enter the maximum expected crew size (variable MAXCR). Enter the maximum expected number of hours (variable MAXHR). This is usually the maximum expected crew size times eight hours. Enter the maximum expected daily accomplishment (variable ACMAX). These values are used for checking for typographical errors in crew size, labor hours, and accomplishment, in a manner similar to the check for maximum expected material quantity described above. The maxhr value is used for checking both regular and overtime hours, but the individual values (not the sum) are checked. Three numerical entries are made on this line.

LINE LN+3 Enter the first month of the analysis, where 1=January, 2=February, etc. (variable IBM). Enter the last two digits of the year of the first month of the analysis period (variable IBY). Enter the last month of the analysis period (variable IEM). Enter the year of the last month of the analysis period (variable IEY). For example, a period of July 1982 through June 1983 would be entered as 7 82 6 83, and a period from November through December 1983 would be entered as 11 83 12 83. The analysis period cannot span more than two calendar years. A total of four numerical entries are made on this line.

LINE LN+4 A total of 10 numbers are entered on this line:

Enter 1 if a deviation analysis is to be conducted; enter 0 if it is not (variable IDEV).

Enter 1 if the analysis is to be done for the Interstate system; enter 0 if it is not (variable INTOUT).

Enter 1 if the analysis is to be done for the Other State Highway system; enter 0 if it is not (variable OSHOUT).

Enter 1 if the analysis is to be done for the Total system; enter 0 if it is not (variable TOTOUT).

Enter 1 if a bar chart of average cost is to be printed; enter 0 if it is not (variable ICOST).

Enter 1 if a bar chart of average labor hours per accomplishment unit is to be printed; enter 0 if it is not (variable IHOURS).

Enter 1 if a bar chart of total period accomplishment is to be printed; enter 0 if it is not (variable ITACC).

Enter 1 if a bar chart of average daily accomplishment is to be printed; enter 0 if it is not (variable IAVACC).

Enter 1 if a bar chart of average crew size is to be printed; enter 0 if it is not (variable ICREW).

Enter n if a bar chart of average quantity of the n-th specified material is to be printed; enter 0 if it is not (variable IMAT). A chart for only one material can be printed per run.

LINE LN+5 Enter the number of standard deviations to be used in the deviation analysis (variable SDEV). This is the only entry on the line. This line is included only if the first entry in the preceding line is 1. Otherwise, LINE LN+4 is the last line of the file.

Figure A.1 is an example parameter input file for activity 201. It specifies six materials, indicates that the analysis is to be done for the Total highway system only, that a deviation analysis is to be done using 1.0 standard deviation, and that all six bar charts are to be printed. The material chart is to be for the 4-th specified material, code number 4431 (bituminous material). The analysis period is from July 1982 through June 1983.

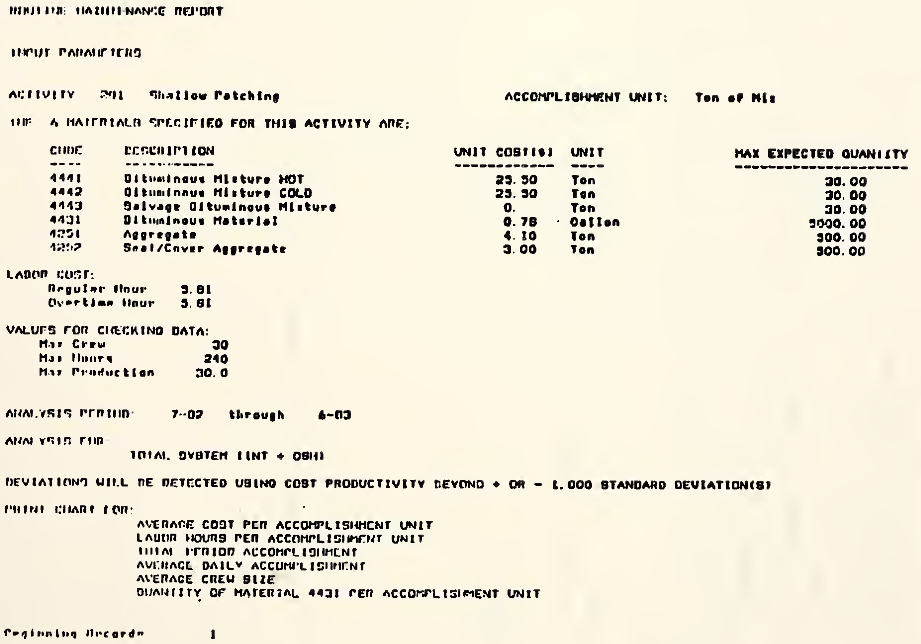
Program Output

The output resulting from the parameter input file of Figure A.I is presented in Figure A.2. The first page echoes the program parameters, describing the analysis that was conducted. If any errors in the parameters were found, appropriate error messages would be listed next. The next pages list possible errors in reading the crew day card data file, or possible errors in the data themselves. The error messages are described in the following section.

The next page printed is an output key, explaining the factors that are presented in the next several pages. The labor factors are presented first, followed by the material, and cost factors. Then the results of the deviation analysis are presented, if such an analysis was performed. Finally, the bar charts are printed. If more than one system were analyzed, similar pages for each system would be printed.

```
1
201 6
Shallow Patching
Ton of Mix
4441 25.5 30.
Bituminous Mixture HOT
Ton
4442 25.5 30.
Bituminous Mixture COLD
Ton
4443 0.0 30.
Salvage Bituminous Mixture
Ton
4431 .78 5000.
Bituminous Material
Gallon
4251 4.1 500.
Aggregate
Ton
4252 3. 500.
Seal/Cover Aggregate
Ton
5.81 5.81
30 240 30.0
7 82 6 83
1 0 0 1    1 1 1 1 1 4
1.
```

Figure A.1. Example of Parameter Input File

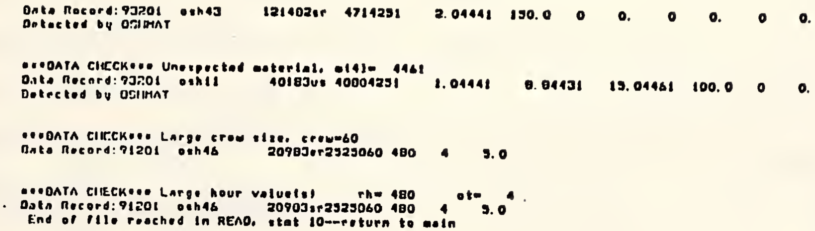


```
ROUTINE MAINTENANCE REPORT

INPUT PARAMETERS

ACTIVITY    201    Shallow Patching                         ACCOMPLISHMENT UNIT:    Ton of Mix

THE  6 MATERIALS SPECIFIED FOR THIS ACTIVITY ARE:

     CODE      DESCRIPTION                                  UNIT COST($)  UNIT           MAX EXPECTED QUANITY
     ----      -----------                                  ------------  ----           --------------------
     4441      Bituminous Mixture HOT                             25.50   Ton                           30.00
     4442      Bituminous Mixture COLD                            25.50   Ton                           30.00
     4443      Salvage Bituminous Mixture                         0.      Ton                           30.00
     4431      Bituminous Material                                0.78  · Gallon                      5000.00
     4251      Aggregate                                          4.10    Ton                          500.00
     4252      Seal/Cover Aggregate                               3.00    Ton                          500.00

LABOR COST:
     Regular Hour     5.81
     Overtime Hour    5.81

VALUES FOR CHECKING DATA:
    Max Crew              30
    Max Hours            240
    Max Production      30.0


ANALYSIS PERIOD      7-82    through    6-83

ANALYSIS FOR
               TOTAL SYSTEM (INT + OSH)

DEVIATIONS WILL BE DETECTED USING COST PRODUCTIVITY BEYOND + OR - 1.000 STANDARD DEVIATION(S)

PRINT CHART FOR:
                AVERAGE COST PER ACCOMPLISHMENT UNIT
                LABOR HOURS PER ACCOMPLISHMENT UNIT
                TOTAL PERIOD ACCOMPLISHMENT
                AVERAGE DAILY ACCOMPLISHMENT
                AVERAGE CREW SIZE
                QUANTITY OF MATERIAL 4431 PER ACCOMPLISHMENT UNIT

Beginning Records       1
```

Figure A.2. Example RMPS Output

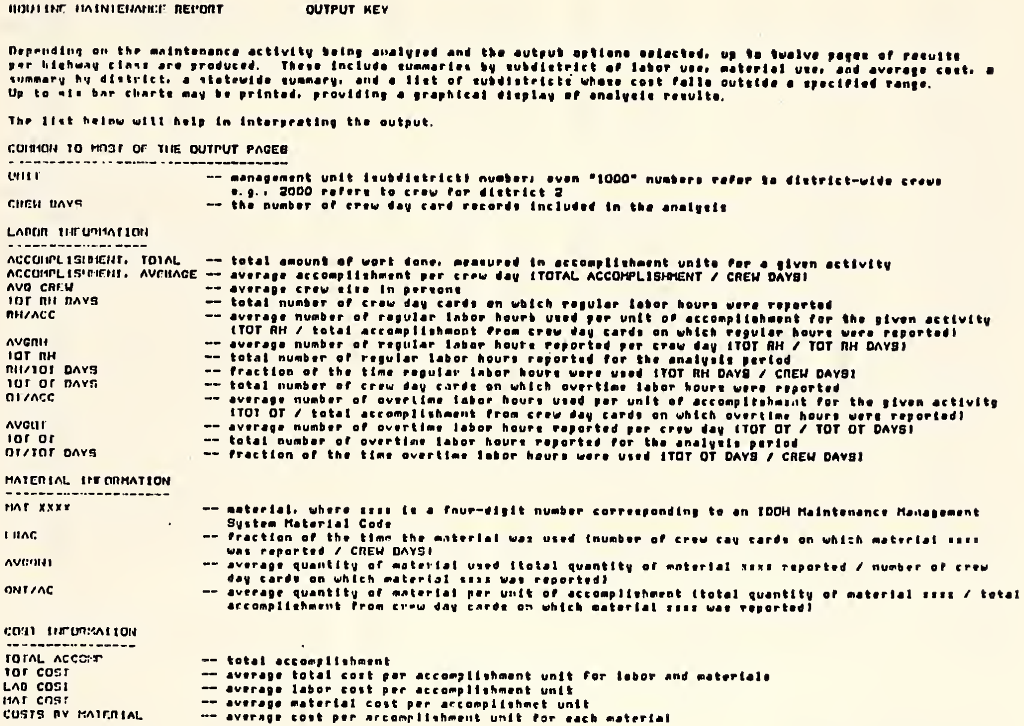
```
***ERROR*** Unexpected card type      TYPE=  00Record examined, IREX=        1
   Error detected by subroutine CKRD1

***ERROR*** Read act code is less than 200 or greater than 299      ACT=    0    Record, IREX=      1
   Error detected by subroutine CKRD1

***ERROR*** Month read is less than 1 or greater than 12      MONTH=   0Record, IREX=     1
   Error detected by subroutine CKRD1

The above 3 error(s) came from this data:00  0                0   0

***DATA CHECK*** Questionable Accomplishment Value      ACC=    40.0
Data Record:91201   osh23       80282sr  3 2 7  48   0   40.0

***DATA CHECK***  LARGE MATERIAL QUANTITY  MAT=  4443  QUAN=      40.0
Data Record:93201   osh23       80282sr  3 24443   40.0   0    0.     0    0.     0    0.     0    0.
Detected by OSHMAT

***DATA CHECK*** Large crew size, crew=60
Data Record:71201   osh47       91682sr1524560  24   0    5.0

***DATA CHECK*** Questionable Accomplishment Value      ACC=    40.0
Data Record:71201   osh33       91302sr  333 7  56   0   40.0

***DATA CHECK*** Unexpected material, m(2)=      1
Data Record:93201   osh52       91402sr 37534441   18.6   1    8.6   0    0.     0    0.     0    0.
Detected by OSHMAT

***DATA CHECK*** Unexpected material, m(1)=  4420
Data Record:93201   int33      110482in 70894420    3.0   0    0.     0    0.     0    0.     0    0.
Detected by INTMAT

***ERROR*** Month read is less then 1 or greater than 12      MONTH=  91Record, IREX=  3121
   Error detected by subroutine CKRD1

The above 1 error(s) came from this data:9m201              91  24

***ERROR*** Month read is less than 1 or greater than 12      MONTH=  92Record, IREX=  3144
   Error detected by subroutine CKRD1

The above 1 error(s) came from this data:9m201              92  24

***DATA CHECK*** Unexpected material, m(1)=  4420
Data Record:73201   osh12      120682sr136 64420    2.0   0    0.     0    0.     0    0.     0    0.
Detected by OSHMAT

***DATA CHECK***  LARGE MATERIAL QUANTITY  MAT=  4441  QUAN=     150.0
```

Figure A.2. Continued



```
Data Record:93201  osh43     121402sr  4714251    2.04441  150.0  0    0.    0    0.    0    0.
Detected by OSIMAT


***DATA CHECK*** Unexpected material, m(4)=  4461
Data Record:93201  osh11      40183us 40004251    1.04441    8.84431   15.04461  100.0  0    0.
Detected by OSIMAT


***DATA CHECK*** Large crew size, crew=60
Data Record:91201  osh46      20983er2525060 480   4    5.0


***DATA CHECK*** Large hour value(s)     rh= 480     ot=   4 .
Data Record:91201  osh46      20903sr2525060 480   4    5.0
 End of file reached in READ, stat 10--return to main
```

Figure A.2. Continued



```
ROUTINE MAINTENANCE REPORT           OUTPUT KEY

Depending on the maintenance activity being analyzed and the output options selected, up to twelve pages of results
per highway class are produced.  These include summaries by subdistrict of labor use, material use, and average cost, a
summary by district, a statewide summary, and a list of subdistricts whose cost falls outside a specified range.
Up to six bar charts may be printed, providing a graphical display of analysis results.

The list below will help in interpreting the output.

COMMON TO MOST OF THE OUTPUT PAGES
----------------------------------
UNIT                      -- management unit (subdistrict) number; even "1000" numbers refer to district-wide crews
                             e.g., 2000 refers to crew for district 2
CREW DAYS                 -- the number of crew day card records included in the analysis

LABOR INFORMATION
-----------------
ACCOMPLISHMENT, TOTAL     -- total amount of wort done, measured in accomplishment units for a given activity
ACCOMPLISHMENT, AVERAGE   -- average accomplishment per crew day (TOTAL ACCOMPLISHMENT / CREW DAYS)
AVG CREW                  -- average crew size in persons
TOT RH DAYS               -- total number of crew day cards on which regular labor hours were reported
RH/ACC                    -- average number of regular labor hourb used per unit of accomplishment for the given activity
                             (TOT RH / total accomplishmont from crew day cards on which regular hours were reported)
AVGRH                     -- average number of regular labor hours reported per crew day (TOT RH / TOT RH DAYS)
TOT RH                    -- total number of regular labor hours reported for the analysis period
RH/TOT DAYS               -- fraction of the time regular labor hours were used (TOT RH DAYS / CREW DAYS)
TOT OT DAYS               -- total number of crew day cards on which overtime labor hours were reported
OT/ACC                    -- average number of overtime labor hours used per unit of accomplishment for the given activitg
                             (TOT OT / total accomplishment from crew day cards on which overtime hours were reported)
AVGOT                     -- average number of overtime labor hours reported per crew day (TOT OT / TOT OT DAYS)
TOT OT                    -- total number of overtime labor hours reported for the analysis period
OT/TOT DAYS               -- fraction of the time overtime labor hours were used (TOT OT DAYS / CREW DAYS)

MATERIAL INFORMATION
--------------------
MAT XXXX                  -- material, where xxxx is a four-digit number corresponding to an IDOH Maintenance Management
                             System Material Code
FMAC                      -- fraction of the time the material was used (number of crew cay cards on which material xxxx
                             was reported / CREW DAYS)
AVGQNT                    -- average quantity of material used (total quantity of material xxxx reported / number of crew
                             day cards on which material xxxx was reported)
QNT/AC                    -- average quantity of material per unit of accomplishment (total quantity of material xxxx / total
                             accomplishment from crew day cards on which material xxxx was reported)

COST INFORMATION
----------------
TOTAL ACCOMP              -- total accomplishment
TOT COST                  -- average total cost per accomplishment unit for labor and materials
LAB COST                  -- average labor cost per accomplishment unit
MAT COST                  -- average material cost per accomplishmet unit
COSTS BY MATERIAL         -- average cost per accomplishment unit for each material
```

Figure A.2. Continued

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                            FROM  7-82 THROUGH  6-83
                      ACCOMPLISHMENT UNIT:       Ton of Mix

TOTAL (INT + OSH) LABOR INFORMATION


         CREW    ACCOMPLISHMENT     AVG      TOT RH                                 RH/TOT   TOT OT                                  OT/TOT
UNIT     DAYS    TOTAL   AVERAGE    CREW      DAYS     RH/ACC    AVG RH    TOT RH    DAYS     DAYS     OT/ACC    AVG OT    TOT OT    DAYS
----    ------   ------  -------   ------    ------    ------    ------    ------    ------   ------   ------    ------    ------   ------


1000        0      0.      0.       0.            0    0.        0.             0    0.            0    0.        0.              0   0.
1100      167    768.7     4.6      6.31        167   10.58     48.70        8133    1.000         3    1.49     11.00            33   0.018
1200      107    369.0     3.4      7.81        108   17.24     58.74        6344    0.991         2    3.33      8.00            16   0.018
1300      161    710.0     4.4      6.29        161   10.85     47.84        7702    1.000         0    0.        0.              0   0.
1400      140    399.5     2.9      6.25        140   16.36     46.69        6536    1.000         0    0.        0.              0   0.
1500      160    485.8     3.0      6.39        160   16.01     48.60        7776    1.000         0    0.        0.              0   0.
1600       84    365.1     4.3      5.87         84   10.33     44.88        3770    1.000         0    0.        0.              0   0.
2000        0      0.      0.       0.            0    0.        0.             0    0.            0    0.        0.              0   0.
2100      228    909.5     4.0      6.34        228   11.61     46.32       10562    1.000         0    0.        0.              0   0.
2200      221   1310.9     5.9      6.29        221    7.96     47.19       10430    1.000         0    0.        0.              0   0.
2300      185    940.5     5.1      6.19        185    8.82     44.83        8294    1.000         0    0.        0.              0   0.
2400      264    933.0     3.5      5.35        264   10.59     37.43        9882    1.000         0    0.        0.              0   0.
2500      283   1043.4     3.7      5.83        283   12.05     44.49       12592    1.000         0    0.        0.              0   0.
2600      178    744.5     4.2      6.15        178   11.35     47.48        8452    1.000         0    0.        0.              0   0.
3000        0      0.      0.       0.            0    0.        0.             0    0.            0    0.        0.              0   0.
3100      231    646.0     2.8      5.87        231   16.74     46.82       10816    1.000         0    0.        0.              0   0.
3200      199    804.5     4.0      6.32        199   12.35     49.91        9932    1.000         0    0.        0.              0   0.
3300      151    630.6     4.2      7.09        151   12.82     53.54        8084    1.000         0    0.        0.              0   0.
3400       87    210.5     2.4      6.46         87   19.62     47.47        4130    1.000         0    0.        0.              0   0.
3500      103    300.0     2.9      5.82        102   15.09     45.80        4672    0.990         1    1.17      7.00             7   0.010
3600      114    545.8     4.8      5.46        114    7.96     38.11        4344    1.000         0    0.        0.              0   0.
4000        0      0.      0.       0.            0    0.        0.             0    0.            0    0.        0.              0   0.
4100      123    489.5     4.0      5.70        123   11.01     43.82        5390    1.000         6    0.25      1.00             6   0.049
4200      206    737.0     3.6      5.44        206   11.10     39.71        8180    1.000         0    0.        0.              0   0.
4300      380   1495.1     3.9      6.06        371   11.73     46.94       17416    0.976        11    3.33      4.64            51   0.029
4400      180    652.7     3.6      5.93        180   12.31     44.64        8036    1.000         0    0.        0.              0   0.
4500      235   1121.4     4.0      5.54        234    9.07     43.37       10148    0.996         3    3.88     10.33            31   0.013
4600      206    920.0     4.5      5.59        206    9.41     42.01        8654    1.000         0    0.        0.              0   0.
4700      213    857.2     4.0      6.00        210   11.10     45.62        9501    0.986         6    3.24      5.67            34   0.028
5000        0      0.      0.       0.            0    0.        0.             0    0.            0    0.        0.              0   0.
5100      165    578.5     3.5      4.52        165    9.95     34.88        5756    1.000         1    0.75      6.00             6   0.006
5200      107    650.6     6.1      4.57        107    5.71     34.71        3714    1.000         0    0.        0.              0   0.
5300      137    402.8     2.9      3.31        137   13.78     40.53        5552    1.000         1    1.33      4.00             4   0.007
5400      172    595.7     3.5      5.09        172   11.12     38.51        6624    1.000         0    0.        0.              0   0.
5500       81    293.7     3.6      3.70         81   11.80     42.80        3467    1.000         0    0.        0.              0   0.
5600      190    928.7     4.9      5.45        190    8.44     41.26        7840    1.000         0    0.        0.              0   0.
6000        0      0.      0.       0.            0    0.        0.             0    0.            0    0.        0.              0   0.
6100      100    283.9     2.8      6.09        100   15.72     44.62        4462    1.000         0    0.        0.              0   0.
6200       74    279.1     3.8      6.03         74   12.76     48.11        3560    1.000         0    0.        0.              0   0.
6300      107    424.4     3.9      5.74        107   11.68     45.47        4956    1.000         0    0.        0.              0   0.
6400       82    293.1     3.6      5.49         82   12.02     42.95        3522    1.000         0    0.        0.              0   0.
6500      132    503.7     3.8      6.28        132   13.00     49.61        6548    1.000         0    0.        0.              0   0.
6600       79    247.0     3.2      5.03         79   12.37     38.84        3068    1.000         0    0.        0.              0   0.
```

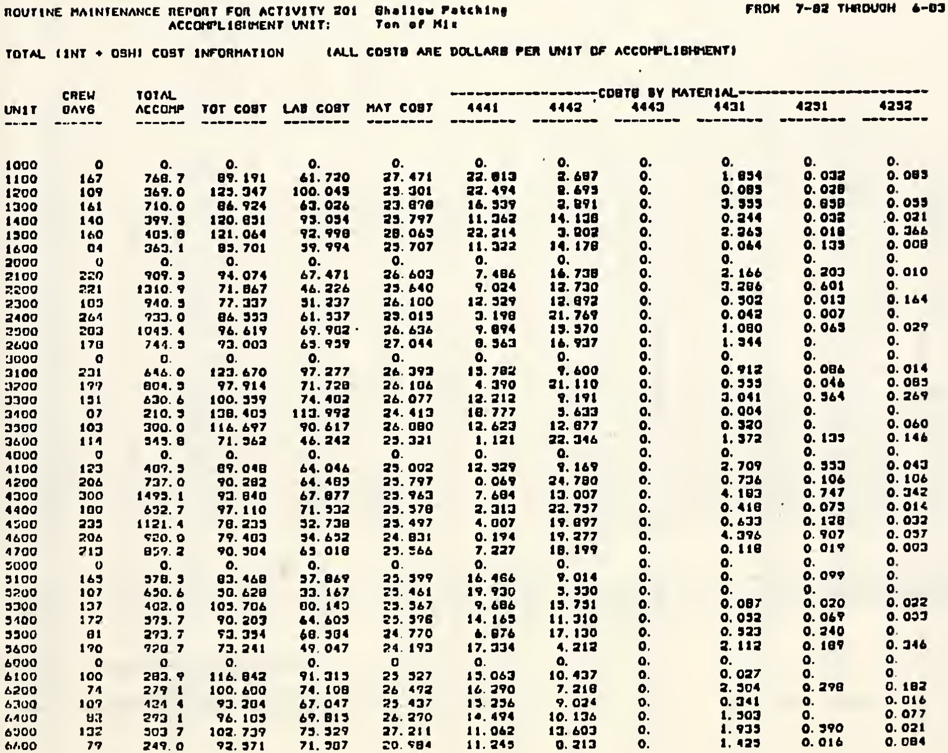
Figure A.2. Continued

ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201 Shallow Patching FROM 7-82 THROUGH 6-83
ACCOMPLISHMENT UNIT: Ton of Mix

TOTAL (INT + OSH) MATERIAL INFORMATION

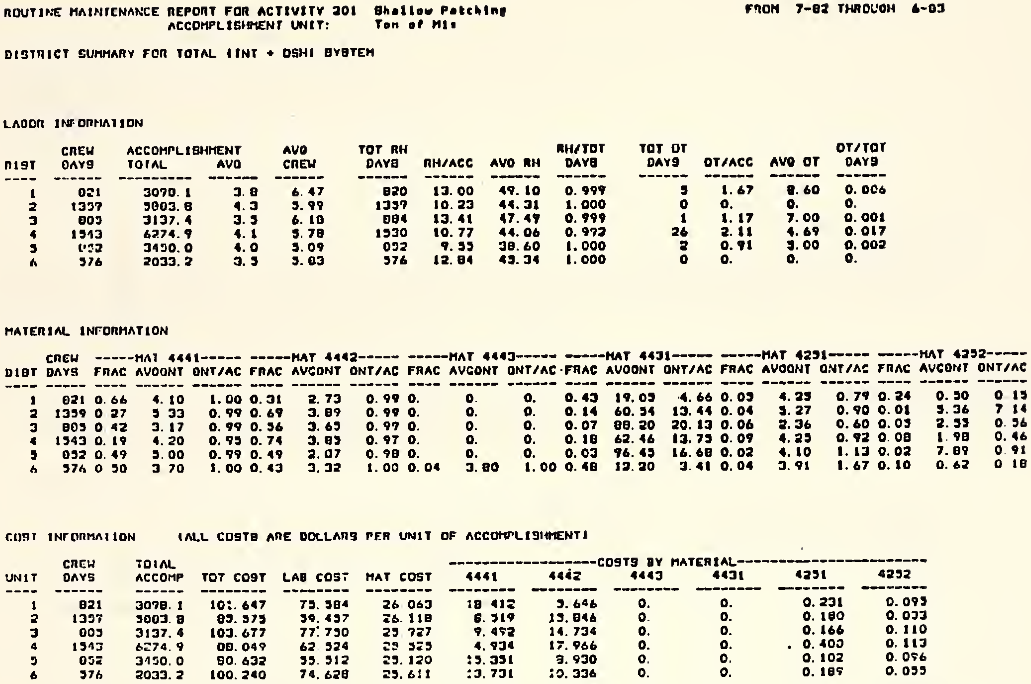
UNIT	CREW DAYS	MAT 4441 FRAC	AVQNT	QNT/AC	MAT 4442 FRAC	AVQNT	QNT/AC	MAT 4443 FRAC	AVQNT	QNT/AC	MAT 4431 FRAC	AVQNT	QNT/AC	MAT 4251 FRAC	AVQNT	QNT/AC	MAT 4252 FRAC	AVQNT	QNT/AC
1000	0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
1100	167	0.87	4.71	1.00	0.13	3.68	1.00	0.	0.	0.	0.86	12.69	2.70	0.03	1.20	0.20	0.17	0.78	0 19
1200	107	0.68	4.40	1.00	0.30	1.18	1.00	0.	0.	0.	0.01	40.00	16.00	0.01	2.30	1.00	0.	0.	0.
1300	161	0.61	4.65	1.00	0.22	2.24	1.00	0.	0.	0.	0.24	82.97	13.15	0.14	6.75	0.95	0.02	3.25	1.00
1400	140	0.41	3.12	0.99	0.61	2.61	0.78	0.	0.	0.	0.07	12.50	4.17	0.05	0.44	0.18	0.05	0.40	0.14
1500	160	0.02	3.21	0.99	0.17	2.26	1.00	0.	0.	0.	0.97	9.10	2.90	0.03	0.42	0.21	0.90	0.38	0.13
1600	84	0.44	4.30	0.98	0.57	4.23	1.00	0.	0.	0.	0.01	30.00	10.00	0.01	12.00	3.00	0.01	1.00	0.33
2000	0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
2100	228	0.25	4.60	0.98	0.71	3.69	0.99	0.	0.	0.	0.35	31.97	7.86	0.04	4.50	1.00	0.01	1.50	0.60
2200	221	0.28	7.60	1.00	0.38	5.11	0.97	0.	0.	0.	0.16	153.42	24.99	0.15	5.82	1.00	0.	0.	0.
2300	185	0.46	5.44	1.00	0.52	4.95	0.99	0.	0.	0.	0.05	67.22	30.25	0.01	3.00	1.00	0.03	10.26	12.82
2400	264	0.14	3.16	0.93	0.87	3.48	0.97	0.	0.	0.	0.01	25.00	14.29	0.00	1.50	1.00	0.	0.	0.
2500	203	0.26	5.40	1.00	0.71	3.18	1.01	0.	0.	.0.	0.10	51.68	11.48	0.01	4.13	0.75	0.02	2.00	******
2600	170	0.30	4.63	1.00	0.70	3.99	1.00	0.	0.	0.	0.81	38.79	8.54	0.	0.	0.	0.	0.	0.
3000	0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
3100	231	0.64	2.70	0.99	0.36	2.90	0.90	0.	0.	0.	0.08	41.94	16.78	0.12	0.50	0.20	0.02	0.75	0.33
3200	177	0.10	3.05	1.00	0.82	4.09	1.00	0.	0.	0.	0.11	27.24	10.40	0.03	1.80	0.42	0.13	0.92	0.30
3300	151	0.39	5.12	0.97	0.51	2.95	0.78	0.	0.	0.	0.10	207.00	23.83	0.12	4.82	0.81	0.07	5.65	0.67
3400	87	0.00	2.21	1.00	0.10	2.91	0 85	0.	0.	0.	0.01	1.00	1.00	0.	0.	0.	0.	0.	0.
3500	103	0.46	3.16	1.00	0.54	2.71	1.00	0.	0.	0.	0.02	100.00	22.22	0.	0.	0.	0.02	3.00	0 67
3600	114	0.07	3.00	0.86	0.88	4.78	0.99	0.	0.	0.	0.07	137.50	24.72	0.04	4.50	1.00	0.04	6.63	1.00
4000	0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
4100	123	0.50	3.94	0.91	0.47	3.03	0 86	0	0.	0.	0.15	89.47	18.68	0.13	4.13	0.97	0.01	7.00	1.00
4200	206	0 00	2.00	1.00	0.97	3.60	1.00	0.	0.	0.	0.03	115.83	36.58	0.03	3.17	1.00	0.03	4.33	1.37
4300	300	0.27	4.13	0.95	0.57	3.53	0.94	0.	0.	0.	0.44	48.30	11.27	0.18	3.95	0.95	0.24	1.85	0.43
4400	180	0.07	4.55	1.00	0.92	3.53	0.99	0.	0.	0.	0.02	116.67	35.00	0.03	2.40	0.51	0.01	3.00	0.75
4500	235	0.18	5.03	0.94	0.82	4.53	0.76	0.	0.	0.	0.09	43.33	9.30	0.04	3.89	0.71	0.05	1.09	0.30
4600	204	0 01	3 30	1.00	0.01	4.19	0 99	0.	0.	0.	0.20	123.45	22.40	0.10	5.35	0.96	0.04	2.19	0.40
4700	213	0.29	3 99	0.99	0.70	4.09	1.00	0.	0.	0.	0.07	3.67	1.73	0.00	4.00	1.00	0.00	1.00	1 00
5000	0	0.	0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5100	165	0.61	3.70	0.99	0.39	3.15	0.97	0.	0.	0.	0.	0.	0.	0.02	4.67	3.33	0.	0.	0.
5200	107	0.62	7.70	1.00	0.38	3.44	0 98	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
5300	137	0.32	2.40	1.00	0.69	2.68	1 00	0.	0.	0.	0.01	22.50	15.00	0.01	2.00	0.67	0.02	1.00	0.60
5400	172	0.42	4.53	0.98	0.60	2.57	0.97	0.	0.	0.	0.03	3.00	2.76	0.01	5.00	2.00	0.01	0.50	0 11
5500	81	0.11	5.90	1.00	0.83	2.94	1 00	0.	0.	0.	0.06	29.40	11.45	0.06	2.44	1.00	0.	0.	0.
5600	173	0.64	5.17	0.99	0.27	2.93	0.78	0.	0.	0.	0.09	147.94	12.91	0.03	4.27	0.91	0.03	10.77	0.95
6000	0	0.	0.	0.	0.	0.	0	0	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.	0.
6100	100	0.50	2.33	1.00	0.30	2.32	1.00	0.	0.	0.	0.01	10.00	5.00	0.	0.	0.	0.	0.	0.
6200	74	0.47	3.09	0.99	0.26	2.93	1.00	0.	0	0.	0.44	17.04	5.92	0.15	4.76	1.00	0.40	0 50	0.21
6300	107	0 62	2.73	1.00	0.22	2.99	1.00	0 01	10.00	1.00	0.22	7.73	2.22	0.	0.	0.	0.05	0 45	0.14
6400	82	0.30	4.06	0.99	0.46	2.07	1.00	0.04	2.50	1.00	0.63	11.46	3.26	0.	0.	0.	0.13	0.65	0.16
6500	132	0 42	2.61	1.00	0.47	4.32	1.00	0 01	6.00	1.00	0.96	7.94	2.36	0.05	7.83	2.27	0.04	0.73	0.14
6600	79	0.41	2.43	1.00	0.33	2.67	1.00	0.22	3.47	1.00	0.34	16.25	3.45	0.01	1.00	0.23	0.07	1.00	0.33

Figure A.2. Continued



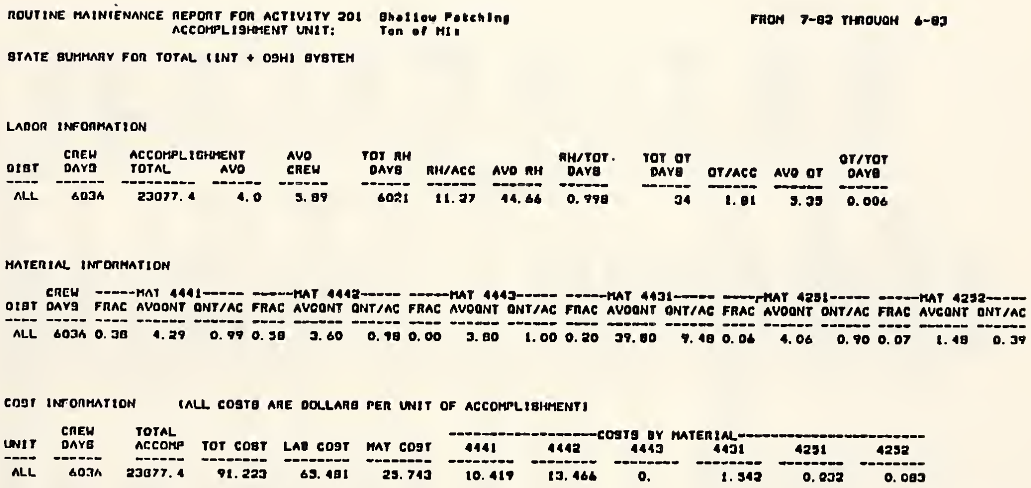
```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                    FROM  7-82 THROUGH  6-83
                   ACCOMPLISHMENT UNIT:      Ton of Mix

TOTAL (INT + OSH) COST INFORMATION     (ALL COSTS ARE DOLLARS PER UNIT OF ACCOMPLISHMENT)

                                                        ------------------COSTS BY MATERIAL---------------------
       CREW      TOTAL
UNIT   DAYS      ACCOMP   TOT COST  LAB COST  MAT COST     4441      4442      4443      4431      4231      4232
----  ------    ------   --------  --------  --------  --------  --------  --------  --------  --------  --------


1000       0        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
1100     167     768.7    89.191    61.720    27.471    22.813     2.687      0.       1.834     0.032     0.083
1200     109     369.0   125.347   100.045    25.301    22.494     2.695      0.       0.085     0.028     0.
1300     161     710.0    86.924    63.026    23.878    16.539     2.891      0.       3.555     0.858     0.035
1400     140     399.5   120.851    95.054    25.797    11.362    14.138      0.       0.244     0.032     0.021
1500     160     405.8   121.064    92.998    28.063    22.214     3.202      0.       2.263     0.018     0.366
1600      84     363.1    85.701    59.994    25.707    11.322    14.178      0.       0.064     0.135     0.008
2000       0        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
2100     220     909.5    94.074    67.471    26.603     7.486    16.738      0.       2.166     0.203     0.010
2200     221    1310.9    71.867    46.226    25.640     9.024    12.730      0.       3.286     0.601     0.
2300     103     940.5    77.337    51.237    26.100    12.529    12.892      0.       0.502     0.013     0.164
2400     264     733.0    86.553    61.537    25.015     3.198    21.769      0.       0.042     0.007     0.
2500     203    1045.4    96.619    69.983    26.636     9.894    15.570      0.       1.080     0.063     0.029
2600     178     744.5    93.003    65.959    27.044     8.563    16.937      0.       1.544     0.        0.
3000       0        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
3100     231     646.0   123.670    97.277    26.393    15.782     9.600      0.       0.912     0.086     0.014
3200     177     804.5    97.914    71.728    26.186     4.390    21.110      0.       0.555     0.046     0.085
3300     151     630.6   100.559    74.482    26.077    12.212     9.191      0.       3.041     0.564     0.269
3400      87     210.5   138.405   113.992    24.413    18.777     5.633      0.       0.004     0.        0.
3500     103     300.0   116.697    90.617    26.080    12.623    12.877      0.       0.520     0.        0.060
3600     114     545.8    71.562    46.242    25.321     1.121    22.346      0.       1.572     0.135     0.146
4000       0        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
4100     123     407.5    89.048    64.046    25.002    12.529     9.169      0.       2.709     0.553     0.043
4200     206     737.0    90.282    64.485    25.797     0.069    24.780      0.       0.736     0.106     0.106
4300     300    1495.1    93.840    67.877    25.963     7.684    13.007      0.       4.183     0.747     0.342
4400     100     652.7    97.110    71.532    25.578     2.313    22.757      0.       0.418     0.075     0.014
4500     235    1121.4    78.235    52.738    25.497     4.807    19.897      0.       0.633     0.128     0.032
4600     206     920.0    79.483    54.652    24.831     0.194    19.277      0.       4.396     0.907     0.057
4700     213     857.2    90.584    65.018    25.566     7.227    18.199      0.       0.118     0.019     0.003
5000       0        0.        0.        0.        0.        0.        0.        0.        0.        0.        0.
5100     165     578.5    83.468    57.869    25.599    16.466     9.014      0.       0.        0.099     0.
5200     107     650.6    58.628    33.167    25.461    19.930     5.530      0.       0.        0.        0.
5300     137     402.0   105.706    80.140    25.567     9.686    15.751      0.       0.087     0.020     0.022
5400     172     575.7    90.203    64.605    25.576    14.165    11.310      0.       0.052     0.069     0.033
5500      81     273.7    93.354    68.584    24.770     6.876    17.130      0.       0.523     0.240     0.
5600     170     720.7    73.241    49.047    24.193    17.334     4.212      0.       2.112     0.189     0.346
6000       0        0         0.        0.        0         0.        0.        0.        0.        0.        0.
6100     100     283.9   116.842    91.315    25.527    15.063    10.437      0.       0.027     0.        0.
6200      74     279.1   100.600    74.108    26.492    16.290     7.218      0.       2.504     0.298     0.182
6300     107     424.4    93.204    67.047    25.437    15.256     9.024      0.       0.341     0.        0.016
6400      83     273.1    96.105    69.815    26.270    14.494    10.136      0.       1.503     0.        0.077
6500     132     503.7   102.739    75.529    27.211    11.062    13.603      0.       1.935     0.590     0.021
6600      77     249.0    92.571    71.587    20.984    11.245     8.213      0.       1.425     0.016     0.084
```

Figure A.2. Continued



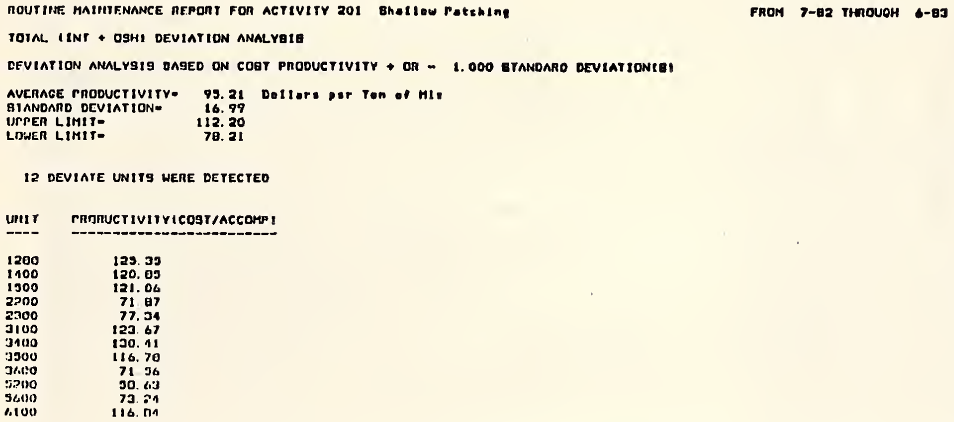
```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                               FROM  7-82 THROUGH  6-83
                    ACCOMPLISHMENT UNIT:     Ton of Mix

DISTRICT SUMMARY FOR TOTAL (INT + OSH) SYSTEM


LABOR INFORMATION

       CREW    ACCOMPLISHMENT      AVG       TOT RH                    RH/TOT     TOT OT                    OT/TOT
DIST   DAYS    TOTAL       AVG     CREW       DAYS    RH/ACC   AVG RH   DAYS       DAYS    OT/ACC  AVG OT   DAYS
----  ------  ---------  ------   ------     ------   ------   ------  ------     ------   ------  ------  ------
   1    821     3078.1     3.8     6.47         820    13.00    49.10   0.999          5     1.67    8.60   0.006
   2   1357     5803.8     4.3     5.99        1357    10.23    44.31   1.000          0     0.      0.      0.
   3    805     3137.4     3.5     6.18         884    13.41    47.49   0.999          1     1.17    7.00   0.001
   4   1543     6274.9     4.1     5.78        1530    10.77    44.06   0.992         26     2.11    4.69   0.017
   5    852     3450.0     4.0     5.09         852     9.55    38.60   1.000          2     0.91    3.00   0.002
   6    576     2033.2     3.5     5.83         576    12.84    43.34   1.000          0     0.      0.      0.


MATERIAL INFORMATION

     CREW  -----MAT 4441-----  -----MAT 4442-----  -----MAT 4443-----  -----MAT 4431-----  -----MAT 4251-----  -----MAT 4252-----
DIST DAYS  FRAC AVGONT ONT/AC FRAC AVGONT ONT/AC FRAC AVGONT ONT/AC FRAC AVGONT ONT/AC FRAC AVGONT ONT/AC FRAC AVGONT ONT/AC
---- ----- ---- ------ ------ ---- ------ ------ ---- ------ ------ ---- ------ ------ ---- ------ ------ ---- ------ ------
   1   821 0.66   4.10   1.00 0.31   2.73   0.99 0.        0.       0. 0.43  19.05   4.66 0.05   4.25   0.79 0.24   0.50   0.15
   2  1359 0.27   5.33   0.99 0.69   3.89   0.99 0.        0.       0. 0.14  60.54  13.44 0.04   5.27   0.90 0.01   5.36   7.14
   3   805 0.42   3.17   0.99 0.56   3.65   0.97 0.        0.       0. 0.07  88.20  20.13 0.06   2.36   0.60 0.05   2.55   0.56
   4  1543 0.19   4.20   0.95 0.74   3.85   0.97 0.        0.       0. 0.18  62.46  13.75 0.09   4.25   0.92 0.08   1.98   0.46
   5   852 0.49   5.00   0.99 0.49   2.07   0.98 0.        0.       0. 0.03  96.45  16.68 0.02   4.10   1.13 0.02   7.89   0.91
   6   576 0.50   3.70   1.00 0.43   3.32   1.00 0.04   3.80   1.00 0.48  12.20   3.41 0.04   3.91   1.67 0.10   0.62   0.18


COST INFORMATION      (ALL COSTS ARE DOLLARS PER UNIT OF ACCOMPLISHMENT)

       CREW     TOTAL                                      -----------------COSTS BY MATERIAL---------------------
UNIT   DAYS     ACCOMP   TOT COST  LAB COST  MAT COST    4441      4442      4443      4431      4251      4252
----  ------    ------   --------  --------  --------  --------  --------  --------  --------  --------  --------
   1    821     3078.1   101.647    75.584    26.063    18.412     5.646    0.        0.          0.231     0.095
   2   1357     5803.8    85.575    59.457    26.118     6.519    13.846    0.        0.          0.180     0.033
   3    805     3137.4   103.677    77.750    25.727     9.492    14.734    0.        0.          0.166     0.110
   4   1543     6274.9    88.049    62.524    25.525     4.934    17.966    0.        0.          0.400     0.113
   5    852     3450.0    80.632    55.512    25.120    15.351     8.930    0.        0.          0.102     0.096
   6    576     2033.2   100.240    74.628    25.611    13.731    10.336    0.        0.          0.189     0.055
```

Figure A.2.   Continued



```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                              FROM  7-82 THROUGH  6-83
                        ACCOMPLISHMENT UNIT:      Ton of Mix

STATE SUMMARY FOR TOTAL (INT + OSH) SYSTEM


LABOR INFORMATION

       CREW    ACCOMPLISHMENT       AVG        TOT RH                    RH/TOT.    TOT OT                    OT/TOT
DIST   DAYS    TOTAL      AVG       CREW        DAYS    RH/ACC   AVG RH   DAYS       DAYS    OT/ACC   AVG OT   DAYS
----  ------  ---------  ------    ------      ------   ------   ------  ------     ------   ------   ------  ------
 ALL    6036    23077.4     4.0      5.89        6021    11.27    44.66   0.998         34     1.81     3.35   0.006


MATERIAL INFORMATION

     CREW  -----MAT 4441----- -----MAT 4442----- -----MAT 4443----- -----MAT 4431----- -----MAT 4251----- -----MAT 4252-----
DIST DAYS  FRAC AVGQNT QNT/AC FRAC AVGQNT QNT/AC FRAC AVGQNT QNT/AC FRAC AVGQNT QNT/AC FRAC AVGQNT QNT/AC FRAC AVGQNT QNT/AC
---- ----- ---- ------ ------ ---- ------ ------ ---- ------ ------ ---- ------ ------ ---- ------ ------ ---- ------ ------
 ALL  6036 0.38   4.29   0.99 0.38   3.60   0.98 0.00   3.80   1.00 0.20  39.80   9.48 0.06   4.06   0.90 0.07   1.48   0.39


COST INFORMATION      (ALL COSTS ARE DOLLARS PER UNIT OF ACCOMPLISHMENT)

       CREW      TOTAL                                      ------------------COSTS BY MATERIAL---------------------
UNIT   DAYS      ACCOMP   TOT COST  LAB COST  MAT COST      4441       4442       4443       4431       4251       4252
----  ------     ------   --------  --------  --------    --------   --------   --------   --------   --------   --------
 ALL    6036    23077.4     91.223    65.481    25.743      10.419     13.466     0.          1.342      0.232      0.083
```

Figure A.2. Continued



```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                    FROM  7-82 THROUGH  6-83

TOTAL (INT + OSH) DEVIATION ANALYSIS

DEVIATION ANALYSIS BASED ON COST PRODUCTIVITY + OR -  1.000 STANDARD DEVIATION(S)

AVERAGE PRODUCTIVITY=    95.21  Dollars per Ton of Mix
STANDARD DEVIATION=      16.99
UPPER LIMIT=            112.20
LOWER LIMIT=             78.21


  12 DEVIATE UNITS WERE DETECTED


UNIT    PRODUCTIVITY(COST/ACCOMP)
----    -------------------------

1200         125.35
1400         120.85
1500         121.06
2200          71.87
2300          77.34
3100         123.67
3400         130.41
3500         116.70
3600          71.36
5200          50.63
5600          73.24
6100         116.04
```

Figure A.2. Continued

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                      FROM  7-82 THROUGH  6-83

TOTAL (INT + OSH) SYSTEM

Average Cost=          95.21  Dollars per Ton of Mix
Standard Deviation=    16.99  Dollars per Ton of Mix

------------------------------------------------------------------------------------------------------------------------------
I                                                                                              I AVERAGE   CREW   ACCOMPLISHMENT  I
I UNIT           Chart Shows Average Cost in Dollars per Ton of Mix                            I  COST     DAYS   TOTAL   AVERAGE I
I ----    -------------------------------------------------------------------------------------I -------  ------ -------- ------- I
I                                   3         2         1         A         1         2        I                                  I
I 1100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I   89.19    167     768.7     4.6 I
I 1200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX          I  123.35    109     369.0     3.4 I
I 1300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I   86.92    161     710.0     4.4 I
I 1400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX             I  120.85    140     399.5     3.9 I
I 1500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX            I  121.06    160     405.0     3.0 I
I 1600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I   85.70     84     365.1     4.3 I
I 2100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I   94.07    228     909.5     4.0 I
I 2200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I   71.87    221    1310.9     5.9 I
I 2300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                       I   77.34    185     940.5     5.1 I
I 2400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I   86.35    264     933.0     3.5 I
I 2500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   96.62    283    1045.4     3.7 I
I 2600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I   93.00    178     744.5     4.2 I
I 3100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX          I  123.67    231     646.0     2.8 I
I 3200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   97.91    199     804.5     4.0 I
I 3300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I  100.56    151     630.6     4.2 I
I 3400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  I  138.41     87     210.5     2.4 I
I 3500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX               I  116.70    103     300.0     2.9 I
I 3600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I   71.56    114     345.8     4.0 I
I 4100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I   89.05    123     489.5     4.0 I
I 4200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I   90.28    206     737.0     3.6 I
I 4300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I   93.84    380    1495.1     3.9 I
I 4400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   97.11    180     652.7     3.6 I
I 4500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                       I   78.23    235    1121.4     4.8 I
I 4600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I   79.48    206     920.0     4.5 I
I 4700    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I   90.58    213     859.2     4.0 I
I 5100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I   83.47    165     578.5     3.5 I
I 5200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                  I   58.63    107     650.6     6.1 I
I 5300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I  105.71    137     402.8     2.9 I
I 5400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I   90.20    172     595.7     3.5 I
I 5500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I   93.35     81     293.7     3.6 I
I 5600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                         I   73.24    190     928.7     4.9 I
I 6100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX               I  116.84    100     283.9     2.8 I
I 6200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I  100.60     74     279.1     3.8 I
I 6300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I   93.28    107     424.4     3.9 I
I 6400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   96.11     82     293.1     3.6 I
I 6500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                       I  102.74    132     503.7     3.8 I
I 6600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I   92.57     79     249.0     3.2 I
I                                   3         2         1         A         1         2        I                                  I
------------------------------------------------------------------------------------------------------------------------------

          A -- Average
          1 -- One Standard Deviation From Average
          2 -- Two Standard Deviations From Average
          3 -- Three Standard Deviations From Average
```

Figure A.2. Continued

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                       FROM  7-82 THROUGH  6-83

TOTAL (INT + OSH) SYSTEM

Average Labor Hours (Reg + Overtime)=   11.97  Hours per Ton of Mix
Standard Deviation=                      2.90  Hours per Ton of Mix

-----------------------------------------------------------------------------------------------------------------------
I
I UNIT     Chart Shows Average Labor Hours (Reg + Overtime) per Ton of Mix            I HOURS/   AVERAGE   ACCOMPLISHMENT  I
I ----   ------------------------------------------------------------------------  I ACCOMP    COST     TOTAL   AVERAGE I
I                    3          2          1          A          1          2      I ------   -------  -------  ------- I
I 1100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                    I  10.62    89.19     768.7     0.    I
I 1200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX           I  17.22   125.35     369.0     0.    I
I 1300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I  10.85    86.92     710.0     0.    I
I 1400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX              I  16.36   130.85     399.5     0.    I
I 1500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX               I  16.01   121.06     483.8     0.    I
I 1600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                     I  10.33    85.70     365.1    89.2   I
I 2100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I  11.61    94.07     909.5    85.7   I
I 2200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                              I   7.96    71.87    1310.9     0.    I
I 2300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                           I   8.82    77.34     940.5     0.    I
I 2400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                     I  10.59    86.55     933.0     0.    I
I 2500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I  12.05    96.62    1045.4     0.    I
I 2600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I  11.35    93.00     744.5    94.1   I
I 3100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX            I  16.74   123.67     646.0    93.0   I
I 3200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I  12.35    97.91     804.5     0.    I
I 3300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I  12.82   100.56     630.6     0.    I
I 3400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  I  19.62   130.41     210.5     0.    I
I 3500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                 I  15.60   116.70     300.0     0.    I
I 3600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                              I   7.96    71.56     545.8   123.7   I
I 4100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I  11.02    89.05     489.5    71.6   I
I 4200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I  11.10    90.28     737.0     0.    I
I 4300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I  11.68    93.84    1495.1     0.    I
I 4400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I  12.31    97.11     632.7     0.    I
I 4500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I   9.08    78.23    1121.4     0.    I
I 4600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I   9.41    79.48     920.0    89.0   I
I 4700   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I  11.19    90.58     859.2    90.3   I
I 5100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I   9.96    83.47     378.5    79.5   I
I 5200   XXXXXXXXXXXXXXXXXXXXXX                                                      I   5.71    50.63     630.6    90.6   I
I 5300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I  13.79   103.71     402.8     0.    I
I 5400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I  11.12    90.20     395.7     0.    I
I 5500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I  11.80    93.35     293.7     0.    I
I 5600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                            I   8.44    73.24     928.7    83.5   I
I 6100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I  15.72   116.84     283.9    73.2   I
I 6200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I  12.76   100.60     279.1     0.    I
I 6300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I  11.68    93.28     424.4     0.    I
I 6400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I  12.02    96.11     293.1     0.    I
I 6500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I  13.00   102.74     503.7     0.    I
I 6600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I  12.32    92.57     249.0   116.8   I
I                    3          2          1          A          1          2      I                                     I
-----------------------------------------------------------------------------------------------------------------------

           A -- Average
           1 -- One Standard Deviation From Average
           2 -- Two Standard Deviations From Average
           3 -- Three Standard Deviations From Average
```

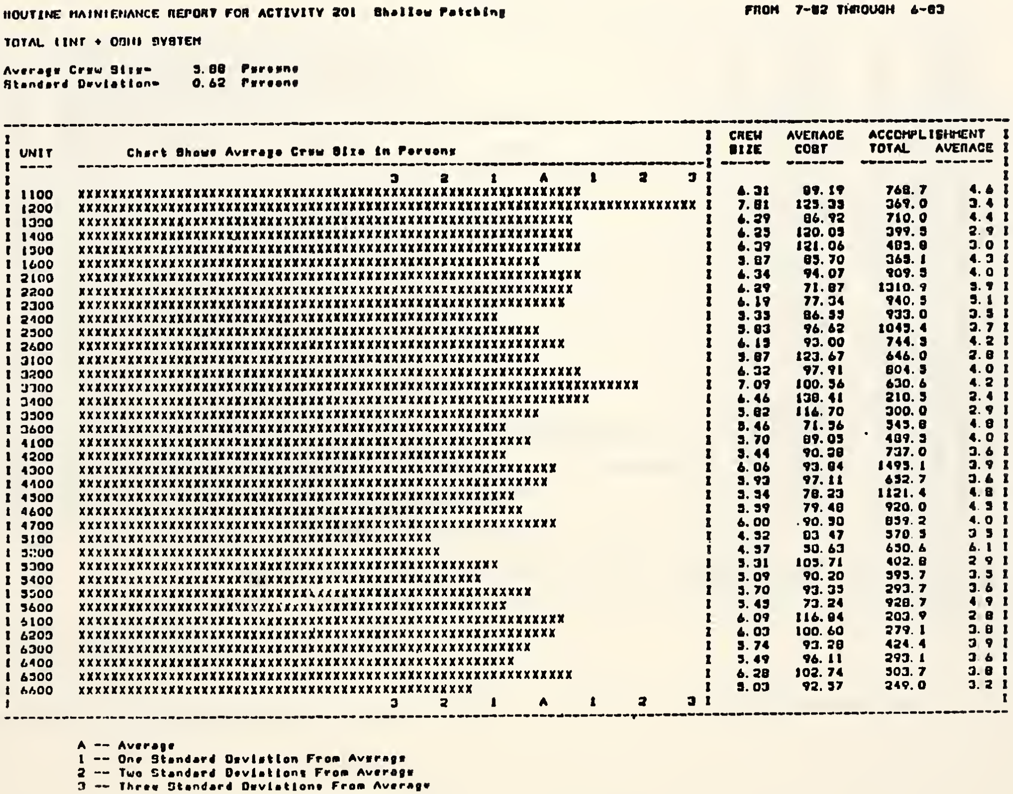
Figure A.2. Continued

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                                   FROM  7-82 THROUGH  6-83

TOTAL (INT + OSHS SYSTEM

Average Total Accomplishment=     645.3  Ton of Mix
Standard Deviation=               309.3  Ton of Mix

-------------------------------------------------------------------------------------------------------------------------------
I                                                                                 I ACCOMPLISHMENT   CREW    AVERAGE  I
I UNIT                  Chart Shows Total Accomplishment In Ton of Mix           I TOTAL  AVERAGE   DAYS    COST     I
I ----    --------------------------------------------------------------------------I -------  -------  ------  -------  I
I         2               1               A               1               2      I                                   I
I 1100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                 I   768.7     4.6     167    89.19  I
I 1200    XXXXXXXXXXXXXXXXXXXX                                                     I   369.0     3.4     109   125.35  I
I 1300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                    I   710.0     4.4     161    86.92  I
I 1400    XXXXXXXXXXXXXXXXXXXXX                                                    I   399.5     2.9     140   120.83  I
I 1500    XXXXXXXXXXXXXXXXXXXXXXXXX                                                I   483.8     3.0     160   121.06  I
I 1600    XXXXXXXXXXXXXXXXXXX                                                      I   365.1     4.3      84    83.70  I
I 2100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   909.5     4.0     228    94.07  I
I 2200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX      I  1310.9     5.9     221    71.87  I
I 2300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I   940.5     5.1     185    77.34  I
I 2400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I   933.0     3.5     264    86.55  I
I 2500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I  1043.4     3.7     203    96.62  I
I 2600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I   744.3     4.2     170    93.00  I
I 3100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I   646.0     2.8     231   123.67  I
I 3200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I   804.5     4.0     199    97.91  I
I 3300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I   630.6     4.2     151   100.56  I
I 3400    XXXXXXXXXXX                                                              I   210.5     2.4      87   130.41  I
I 3500    XXXXXXXXXXXXXXX                                                          I   300.0     2.9     103   116.70  I
I 3600    XXXXXXXXXXXXXXXXXXXXXXXXXXXX                                             I   543.8     4.8     114    71.56  I
I 4100    XXXXXXXXXXXXXXXXXXXXXXXXXX                                               I   489.5     4.0     123    89.05  I
I 4200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I   737.0     3.6     206    90.28  I
I 4300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX I  1493.1     3.9     380    93.84  I
I 4400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                       I   652.7     3.6     180    97.11  I
I 4500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I  1121.4     4.8     235    78.23  I
I 4600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   920.0     4.5     206    79.40  I
I 4700    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I   839.2     4.0     213    90.58  I
I 5100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                           I   570.5     3.5     165    83.47  I
I 5200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                       I   650.6     6.1     107    50.63  I
I 5300    XXXXXXXXXXXXXXXXXXXXX                                                    I   402.8     2.9     137   105.71  I
I 5400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                          I   593.7     3.5     172    90.20  I
I 5500    XXXXXXXXXXXXXXX                                                          I   293.7     3.6      81    93.35  I
I 5600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I   928.7     4.9     190    73 24  I
I 6100    XXXXXXXXXXXXXX                                                           I   283.9     2.8     100   116.84  I
I 6200    XXXXXXXXXXXXXX                                                           I   279.1     3.8      74   100.60  I
I 6300    XXXXXXXXXXXXXXXXXXXXXX                                                   I   424.4     3.9     109    93 20  I
I 6400    XXXXXXXXXXXXXXX                                                          I   293.1     3.6      82    96.11  I
I 6500    XXXXXXXXXXXXXXXXXXXXXXXXXX                                               I   503.7     3.0     132   102 74  I
I 6600    XXXXXXXXXXXX                                                             I   249.0     3.2      79    97.57  I
I         2               1               A               1               2      I                                   I
-------------------------------------------------------------------------------------------------------------------------------

          A -- Average
          1 -- One Standard Deviation From Average
          2 -- Two Standard Deviations From Average
          3 -- Three Standard Deviations From Average
```

Figure A.2. Continued

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                    FROM  7-82 THROUGH  6-83

TOTAL (INI + OSH) SYSTEM

Average Daily Accomplishment=        3.9  Ton of Mix
Standard Deviation=                  0.8  Ton of Mix

----------------------------------------------------------------------------------------------------------------
I                                                                        I  ACCOMPLISHMENT     CREW   AVERAGE I
I UNIT       Chart Shows Average Daily Accomplishment in Ton of Mix      I AVERAGE   TOTAL     DAYS    COST   I
I ----   ----------------------------------------------------------------I -------  -------   -----  ------- I
I                      3       2       1       A       1       2         I                                    I
I 1100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I   4.6     768.7     167    09.19  I
I 1200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                              I   3.4     369.0     109   125.35  I
I 1300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I   4.4     710.0     161    86.92  I
I 1400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I   2.9     399.5     140   120.85  I
I 1500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                  I   3.0     485.8     160   121.06  I
I 1600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I   4.3     365.1      84    85.70  I
I 2100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I   4.0     909.5     228    94.07  I
I 2200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX     I   5.9    1310.9     221    71.87  I
I 2300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX             I   5.1     940.5     185    77.34  I
I 2400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I   3.5     933.0     264    06.55  I
I 2500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I   3.7    1045.4     203    96.62  I
I 2600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I   4.2     744.5     178    93.00  I
I 3100   XXXXXXXXXXXXXXXXXXXXXXXXXXXX                                    I   2.8     646.0     231   123.67  I
I 3200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I   4.0     804.9     199    97.91  I
I 3300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I   4.2     630.6     151   100.56  I
I 3400   XXXXXXXXXXXXXXXXXXXXXXXX                                        I   2.4     210.9      87   130.41  I
I 3500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I   2.9     300.0     103   116.70  I
I 3600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I   4.8     545.8     114    71.56  I
I 4100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I   4.0     489.5     123    89.05  I
I 4200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   3.6     737.0     206    90.28  I
I 4300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX         .               I   3.9    1495.1     300    93.84  I
I 4400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   3.6     652.7     180    97.11  I
I 4500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I   4.8    1121.4     235    78 23  I
I 4600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   I   4.5     920.0     206    79.48  I
I 4700   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I   4.0     859.2     213    90 58  I
I 5100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I   3.5     570.5     165    03.47  I
I 5200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX   I   6.1     630.6     107    50.63  I
I 5300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                   I   2.9     402.8     137   105.71  I
I 5400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I  .3.5     595.7     172    90.20  I
I 5500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   3.6     293.7      81    93 35  I
I 5600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX               I   4.9     928.7     190    73.24  I
I 6100   XXXXXXXXXXXXXXXXXXXXXXXXXXXX                                    I   2.8     283.9     100   116.84  I
I 6200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   3.8     279.1      74   100.60  I
I 6300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I   3.9     424.4     109    93.28  I
I 6400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I   3.6     293.1      82    96.11  I
I 6500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I   3.8     303.7     132   102.74  I
I 4600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I   3.2     249.0      77    92.57  I
I                      3       2       1       A       1       2         I                                    I
----------------------------------------------------------------------------------------------------------------

         A -- Average
         1 -- One Standard Deviation From Average
         2 -- Two Standard Deviations From Average
         3 -- Three Standard Deviations From Average
```

Figure A.2. Continued



```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                          FROM  7-82 THROUGH  6-83

TOTAL (INT + OOH) SYSTEM

Average Crew Size=      5.88  Persons
Standard Deviation=     0.62  Persons

---------------------------------------------------------------------------------------------------------------------------------------------
I                                                                                                  I  CREW    AVERAGE    ACCOMPLISHMENT     I
I UNIT         Chart Shows Average Crew Size in Persons                                            I  SIZE     COST      TOTAL    AVERAGE   I
I ----    -----------------------------------------------------------------------------------------I  ------  -------   --------  -------   I
I                                                     3      2      1      A      1      2      3  I                                        I
I 1100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I    6.31     89.19       768.7      4.6 I
I 1200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX I    7.81    125.33       369.0      3.4 I
I 1300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I    6.29     86.92       710.0      4.4 I
I 1400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I    6.25    120.05       399.5      2.9 I
I 1500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                 I    6.39    121.06       485.8      3.0 I
I 1600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                       I    5.87     85.70       365.1      4.3 I
I 2100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                 I    6.34     94.07       909.5      4.0 I
I 2200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I    6.29     71.87      1310.9      5.9 I
I 2300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                   I    6.19     77.34       940.5      5.1 I
I 2400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I    5.35     86.55       933.0      3.5 I
I 2500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                       I    5.83     96.62      1045.4      3.7 I
I 2600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I    6.15     93.00       744.3      4.2 I
I 3100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                       I    5.87    123.67       646.0      2.8 I
I 3200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I    6.32     97.91       804.5      4.0 I
I 3300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX         I    7.09    100.56       630.6      4.2 I
I 3400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I    6.46    138.41       210.5      2.4 I
I 3500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                       I    5.82    116.70       300.0      2.9 I
I 3600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I    5.46     71.56       345.8      4.8 I
I 4100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I    5.70     89.05       489.3      4.0 I
I 4200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                            I    5.44     90.38       737.0      3.6 I
I 4300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I    6.06     93.84      1495.1      3.9 I
I 4400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                      I    5.93     97.11       652.7      3.6 I
I 4500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I    5.34     78.23      1121.4      4.8 I
I 4600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                          I    5.59     79.48       920.0      4.5 I
I 4700    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I    6.00    .90.50       859.2      4.0 I
I 5100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                      I    4.52     83 47       570.5      3 5 I
I 5200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                     I    4.57     50.63       650.6      6.1 I
I 5300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I    5.31    105.71       402.8      2 9 I
I 5400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                               I    5.09     90.20       595.7      3.5 I
I 5500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                         I    5.70     93.35       293.7      3.6 I
I 5600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I    5.45     73.24       928.7      4 9 I
I 6100    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                    I    6.09    116.84       203.9      2 8 I
I 6200    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                     I    6.03    100.60       279.1      3.8 I
I 6300    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                        I    5.74     93.28       424.4      3 9 I
I 6400    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I    5.49     96.11       293.1      3 6 I
I 6500    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                  I    6.28    102.74       503.7      3.8 I
I 6600    XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I    5.03     92.57       249.0      3.2 I
I                                                     3      2      1      A      1      2      3  I                                        I
---------------------------------------------------------------------------------------------------------------------------------------------

          A -- Average
          1 -- One Standard Deviation From Average
          2 -- Two Standard Deviations From Average
          3 -- Three Standard Deviations From Average
```

Figure A.2. Continued

```
ROUTINE MAINTENANCE REPORT FOR ACTIVITY 201  Shallow Patching                              FROM  7-82 THROUGH  6-83

TOTAL (INT + OSHI SYSTEM

Average Quantity of Material 4431=  12.94 Gallon               per Ton of Mix
Standard Deviation=                  9.76 Gallon               per Ton of Mix

-----------------------------------------------------------------------------------------------------------------------------
I                                                                                      I QUAN/    AVERAGE   ACCOMPLISHMENT  I
I UNIT   Chart Shows Average Quantity of Material 4431 per Accomplishment Unit         I ACCOMP   COST       TOTAL  AVERAGE I
I ----   ------------------------------------------------------------------------------I ------   -------   -------  ------- I
I              1                   A                    1                   2          I                                    I
I 1100   XXXXXX                                                                        I   2.70     89.19     768.7      4.6 I
I 1200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                             I  16.00    125.33     369.0      3.4 I
I 1300   XXXXXXXXXXXXXXXXXXXXXXXXXXX                                                   I  13.15     86.92     710.0      4.4 I
I 1400   XXXXXXXXX                                                                     I   4.17    120.85     399.3      2.9 I
I 1500   XXXXXX                                                                        I   2.98    121.06     483.8      3.0 I
I 1600   XXXXXXXXXXXXXXXXXXXXX                                                         I  10.00     85.70     365.1      4.3 I
I 2100   XXXXXXXXXXXXXXXX                                                              I   7.86     94.07     909.3      4.0 I
I 2200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I  24.99     71.87    1310.9      5.9 I
I 2300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                I  30.23     77.34     940.5      5.1 I
I 2400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                                 I  14.29     86.35     933.0      3.5 I
I 2500   XXXXXXXXXXXXXXXXXXXXXXXX                                                      I  11.48     96.62    1045.4      3.7 I
I 2600   XXXXXXXXXXXXXXXXXX                                                            I   8.54     93.00     744.5      4.2 I
I 3100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                            I  16.78    123.67     646.0      2.8 I
I 3200   XXXXXXXXXXXXXXXXXXXXX                                                         I  10.40     97.91     804.5      4.0 I
I 3300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                             I  23.83    100.56     630.6      4.2 I
I 3400   XX                                                                            I   1.00    130.41     210.5      2.4 I
I 3500   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I  22.22    116.70     300.0      2.9 I
I 3600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                           I  24.72     71.56     543.8      4.8 I
I 4100   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                        I  18.68     89.03     409.5      4.0 I
I 4200   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX  I  36.58     90.20     737.0      3.6 I
I 4300   XXXXXXXXXXXXXXXXXXXXXXX                                                       I  11.27     93.84    1495.1      3.9 I
I 4400   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX     I  35.00     97.11     632.7      3.6 I
I 4500   XXXXXXXXXXXXXXXXXXX                                                           I   9.38     78.23    1121.4      4 8 I
I 4600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                I  22.40     79.48     920.0      4.5 I
I 4700   XXXX                                                                          I   1.73     90.50     839.2      4.0 I
I 5100                                                                                 I   0.       83.47     570.5      3.5 I
I 5200                                                                                 I   0.       58.63     650.6      6.1 I
I 5300   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                               I  15.00    105.71     402.8      2.9 I
I 5400   XXXXXX                                                                        I   2.76     90.20     595.7      3.3 I
I 5500   XXXXXXXXXXXXXXXXXXXXXXX                                                       I  11.43     93.35     293.7      3.6 I
I 5600   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX                                       I  18.91     73.24     928.7      4.9 I
I 6100   XXXXXXXXXX                                                                    I   5.00    116.84     283.9      2.8 I
I 6200   XXXXXXXXXXXX                                                                  I   5.92    100.60     279.1      3.8 I
I 6300   XXXXX                                                                         I   2.22     93.28     424.4      3.9 I
I 6400   XXXXXXX                                                                       I   3.26     96.11     293.1      3.6 I
I 6500   XXXXX                                                                         I   2.54    102.74     303.7      3.8 I
I 6600   XXXXXXXXXXX                                                                   I   5.48     92.57     249.0      3.2 I
I              1                   A                    1                   2          I                                    I
-----------------------------------------------------------------------------------------------------------------------------

         A -- Average
         1 -- One Standard Deviation From Average
         2 -- Two Standard Deviations From Average
         3 -- Three Standard Deviations From Average
```

Figure A.2. Continued

Error and Data Check Messages

Several error or data check messages may be printed by RMPS. An error message is printed in response to some kind of error detected, while a data check message refers to a questionable piece of data. The data may or may not be correct, something that must be determined by the user. This section of the manual describes each possible message, its source, probable cause and recommended corrective action.

Error messages generally have the form, "***ERROR*** 'explanation message'", where 'explanation message' is a brief description of the error. A line of data follows the 'explanation message' in some cases. This section lists the 'explanation messages' from error statements in alphabetical order, notes which subroutine printed the error, and what corrective action should be taken.

'ABNORMAL TERMINATION, End of file reached before ib, Program terminated within subroutine BEGIN.' Printed by subroutine BEGIN. In this case, the value of ib, the first data record to be examined by the program is larger than the total number of lines in the data file. The user is probably using the wrong data file, or he does not know the correct location of the data of interest within the data file. Check data file or set ib to a lower value.

'All material codes are not within correct numerical range.' Printed by subroutine CKPRAM. A material code has been read that is not within the range of 4060-4522, non-inclusive. Either the material code number is incorrect, or more material codes have been added to the IDOH Maintenance Management System. Check the material code number. If a new code has been added, the values in CKPRAM must be changed. (They are set between statements 30 and 40 in CKPRAM.)

'Beginning date is before ending date.' Printed by CKPRAM. The beginning date for the analysis period is earlier than the ending date. Check the dates in the parameter input file.

'Beginning month is less than 1 or greater than 12.' Printed by CKPRAM. Check specification for beginning month in parameter input file. 1=January, 2=February, etc.

'Ending month is less than 1 or greater than 12.' Printed by CKPRAM. Check specification for ending month in parameter input file.

'Ending year is before beginning year.' Printed by CKPRAM. Check specifications for beginning and ending years in parameter input file.

'Ending year is later than beginning year + 1.' Printed by CKPRAM. Check year specifications in parameter file. The analysis period cannot span more than two consecutive years.

'Error reading data in RDLAB stmt 900.' Printed by RDLAB. The current data record is also printed according to the format in Table A.1. A reading error has occurred while executing the read statement just before statement 900 in RDLAB. Likely cause is a format error. Check the data format.

'Error reading data in RDMAT stmt 900.' Printed by RDMAT. The current data record is also printed according to the format in Table A.2. A reading error has occurred while executing the read statement just before statement 900 in RDMAT. Likely cause is a format error. Check the data format.

'Error reading month, yr in CKRD2 stmt 61.' Printed by CKRD2. The values of month and yr are also printed. A reading error occurred in CKRD2 while executing the read statement immediately following statement 61. Likely cause is a format error. Check the data format.

'Error reading month, yr in CKRD2 stmt 141.' Printed by CKRD2. The values of month and yr are also printed. A reading error occurred in CKRD2 while executing the read statement immediately following statement 141. Likely cause is a format error. Check the data format.

'Error reading type, act, month, yr in READ, stmt 10.' Printed by READ. A reading error occurred while executing the read statement, statement 10 in READ. Likely cause is a format error. Check the data format.

'Error reading yr in CKRD2 stmt 50.' Printed by CKRD2. The value of yr is also printed. A reading error occurred in CKRD2 while executing the read statement immediately after statement 50. Likely cause is a format error. Check data format.

'Error reading yr in CKRD2 stmt 131.' Printed by CKRD2. The value of yr is also printed. A reading error occurred in CKRD2 while executing the read statement immediately after statement 131. Likely cause is a format error. Check data format.

'Month read is less than 1 or greater than 12.' Printed by CKRD1. Values of month and irex are also printed. The error was detected on the irex-th record examined. Likely cause is incorrect format on the record. Could also indicate incorrect read format in program.

'More than 6 materials are specified.' Printed by CKPRAM. Value of nmat, the number of specified materials, is greater than 6. Check the value of nmat in the parameter input file.

'Read act code is less than 200 or greater than 299.' Printed by CKRD1. Values of act and irex are also printed. The error was detected on the irex-th record. The activity code was not a "200" number. Likely cause is typographical error on the record. Could also indicate incorrect read format in program. NOTE: If the data are not sorted according to activity, activity codes other than "200" numbers may be read where there is no error.

'Specified activity code is less than 200 or greater than 299.' Printed by CKPRAM. Only activities with "200" codes are anticipated to be analyzed. The specified activity has a code number outside this range. Check the value of iact in the parameter input file. NOTE: If activities with code numbers other than "200" numbers are to be analyzed, this error and the previous error checks must be changed.

'Unexpected card type.' Printed by CKRD1. Values of type and irex are also printed. The error was detected on the irex-th record examined. A card type other than 91, 91, 93, or 9m was read. Likely cause is a typographical error. Could also indicate incorrect read format in program.

'Value of iavacc is incorrect, iavacc= .' Printed by CKPRAM. Iavacc must be 0 or 1. Check value in parameter input file.

'Value of icost is incorrect, icost= .' Printed by CKPRAM. Icost must be 0 or 1. Check value in parameter input file.

'Value of icrew is incorrect, icrew= .' Printed by CKPRAM. Icrew must be 0 or 1. Check value in parameter input file.

'Value of idev is incorrect, idev= .' Printed by CKPRAM. Idev must be 0 or 1. Check value in parameter input file.

'Value of ihours is incorrect, ihours= .' Printed by CKPRAM. Ihours must be 0 or 1. Check value in parameter input file.

'Value of imat is incorrect, imat= .' Printed by CKPRAM. Imat must be 0 or 1. Check value in parameter input file.

'Value of intout is incorrect, intout= .' Printed by CKPRAM. Intout must be 0 or 1. Check value in parameter input file.

'Value of itacc is incorrect, itacc= .' Printed by CKPRAM. Itacc must be 0 or 1. Check value in parameter input file.

'Value of oshout is incorrect, oshout= .' Printed by CKPRAM. Oshout must be 0 or 1. Check value in parameter input file.

'Value of totout is incorrect, totout= .' Printed by CKPRAM. Totout must be 0 or 1. Check value in parameter input file.

A data check message is printed when a value of a variable read from a crew day card record is suspected of being incorrect. The messages are of the general form, "***DATA CHECK*** 'message'," where 'message' explains the questionable data value. The entire data record on which the questionable value appears is also printed. Values read from records which trigger data check messages are not included in the tabulation. The messages are listed below in alphabetical order.

'Large crew size, crew= .' The entire data record is also printed, using the format in table A.1. Printed by RDLAB. The value of crew is larger than the maximum expected crew size, maxcr, specified on line LN+2 of the parameter input file. The data record can be examined to determine if the value is likely in error, or if it appears to be correct. If several of these messages are printed for values of crew that appear to be correct, the value of maxcr should be increased.

'Large hour value(s), rh= , ot= .' Rh is the value of regular hours from the data record, and ot is the value of overtime. The entire data record is also printed, using the format in Table A.1. Printed by RDLAB. The value of rh or ot is larger than the maximum expected hour value, maxhr, specified on line LN+2 of the parameter input file. The data record can be examined to determine if the value is is likely in error, or if it appears to be correct. If several of these messages are printed for values of rh or ot that appear to be correct, the value of maxhr should be increased.

'Large material quantity, mat= , quan= .' Mat is the material code, and quan is the material quantity from the data record. The entire data record is also printed, using the format in table A.2. Printed by INTMAT or OSHMAT. The material quantity read is larger than the maximum expected quantity for that material, qmax, specified in lines 5 through LN of the parameter input file. The data record can be examined to determine if the material quantity is likely a typographical error or if it appears to be correct. If several of these messages are printed for a material, the qmax value may need to be increased.

'Questionable accomplishment value, acc= .' Acc is the accomplishment value read from the data record. The entire data record is also printed. Printed by RDLAB. The accomplishment value read is greater than the expected maximum, acmax, specified on line LN+2 in the input parameter file. The data record can be examined to determine if the value is likely a typographical error, or if it appears to be correct. If several of these messages are printed for accomplishment values that appear to be correct, the value of acmax may need to be increased.

'Unexpected material, m(k)= .' M(k) is the unrecognized material code; it is the k-th material code listed on the data record. The entire data record is also printed according to the format in Table A.2. Printed by INTMAT or OSHMAT. The indicated material code was not one of the specified materials listed in lines 5 through LN of the parameter input file. The record can be examined to determine if there was an apparent coding error. If several of these messages are printed for the same material, and there appears to be no coding error, this material may need to be added to the list of specified materials in the parameter file.

'Unusual class designation, class= .' Class is the roadway class designation read from the record. The entire record is also printed according to the format in Table A.1 if it is a labor record, or Table A.2 if it is a material record. Printed by RDLAB or RDMAT. The roadway class designation is not 'int' or 'osh'. The likely cause is a typographical error.

Program Calculation Procedure and Structure

Language and Resource Requirements

RMPS is written in standard Fortran IV, except for two minor exceptions as explained below, and uses standard library functions. The code is approximately 5500 lines long, including comments.

Because the program was developed on a computer system on which Fortran 77, not Fortran IV was available, character variables had to be declared as such. Thus, if RMPS is to be used on a system with only Fortran IV, the character declaration statements must be removed. These statements occur at the beginning of the main program, MAIN, and all subroutines except the following: BEGIN, CKRD2, INTLAB, OSHLAB, OUTKEY, DSTLAB, DSTMAT, DSTCST, and STDEV. The logical files are assigned with open statements located near the beginning of the main program, MAIN. Three files are used by RMPS. Unit 1 is the parameter input file; unit 2 is the crew card data file, and unit 3 is the output file.

RMPS was developed and tested on a Digital Equipment Corporation VAX 11/780 with a UNIX 4.2(bsd) operating system. The executable code requires 111616 decimal bytes; the initialized data portions require 80896 decimal bytes; and the uninitialized data areas require 21312 bytes, for a total of 213824 decimal bytes. The execution time varies with the program options selected and the size of data file being used. Typical execution times (CPU times) range from one or two minutes for a short analysis period (one or two months) when the data of interest are at the beginning of the file, to approximately ten minutes for an analysis period of a year with activity 201 (the largest activity), when the data file has been sorted to contain only records of that activity for that year.

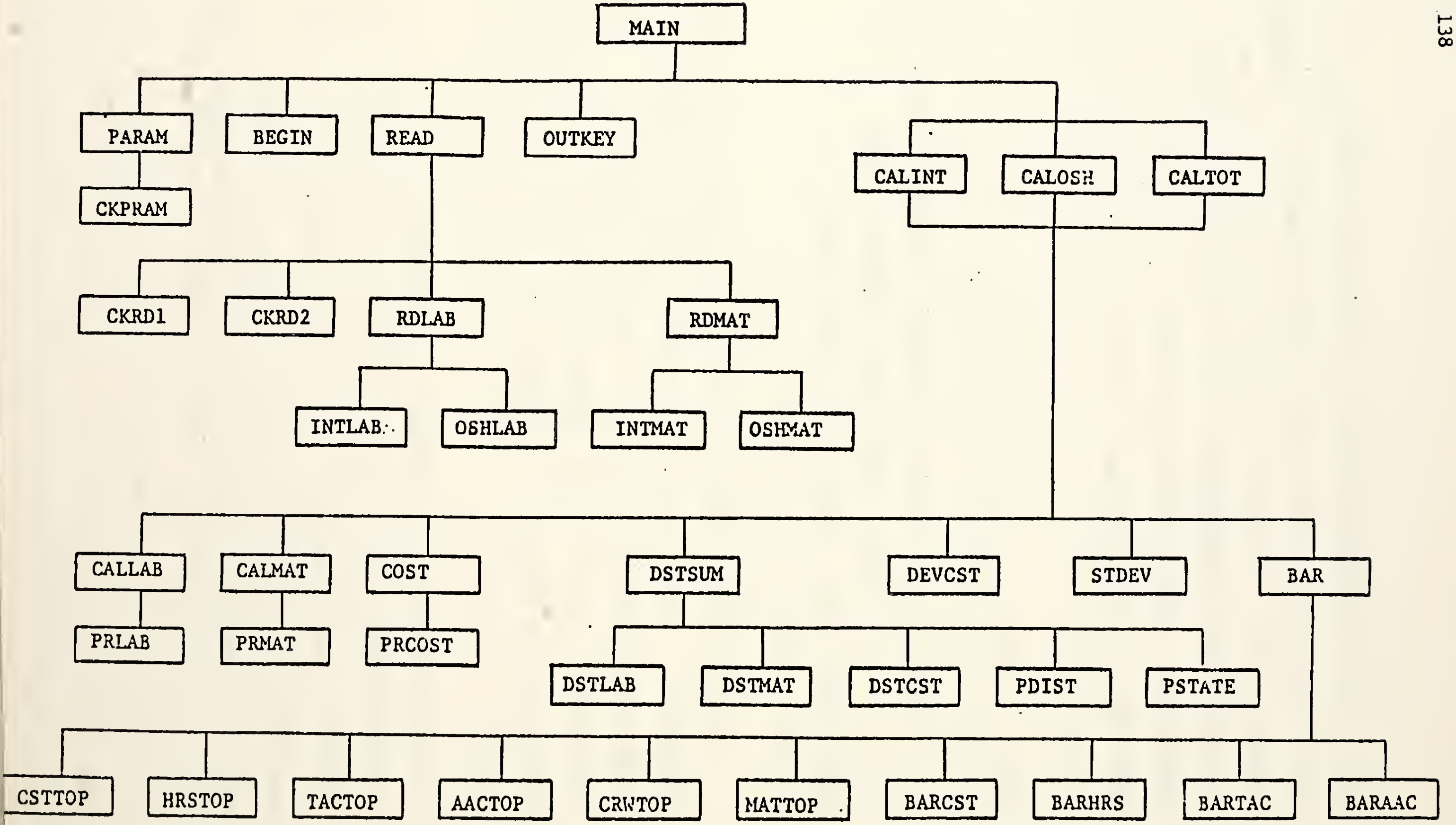
Structure and Calculation Procedure

The program consists of one main program, called MAIN, and 41 subroutines. RMPS has two main sections: one that reads data, and one that analyzes the data. Figure A.3 is a block diagram of the program.

Subroutine MAIN controls the switching between the main sections of the program. Initially, MAIN calls subroutine PARAM which reads the program control parameters. These parameters indicate which activity to analyze, what highway systems the analysis is for, and what type of output is desired.

Subroutine PARAM calls subroutine CKPRAM which echos the parameters, and checks them for reasonableness. The following checks are made, and if any parameter does not have an acceptable value, an error message is printed.

Activity Variable iact specifies the IDOH Maintenance code for the activity being analyzed. The routine



igure A.3. Block Diagram of Program RMPS maintenance activity codes are all numbers from 200 through 299. The activity code must fall within this range.

Number A maximum of six materials may be specified for each activity.

Material Material codes must fall within the range of 4060-4522, non-inclusive. This range was determined from the list of materials in the 1982-83 Field Operations Handbook for Foremen, Division of Maintenance.

Date The analysis period dates are checked to insure that the beginning date is earlier than the ending date, and that month specifications are from 1 through 12 (January -- December). An analysis period cannot span more than two calendar years.

Deviation Variable idev equals 0 if no deviation analysis is to be performed; it equals 1 otherwise.

Highway Variables intout, oshout, totout are 0 if the analysis is not to be done for the interstate, other state highway, or total system, respectively; they are 1 otherwise.

Bar One variable for each type of bar chart indicates if that chart is to be printed. The variable is 0 if the chart is not to be printed; it is one otherwise. The variables are icost, ihours, imat, itacc, iavacc, and icrew for average cost, labor hours, material quantity, total accomplishment, average accomplishment, and crew size, respectively.

If any of the parameters has an unacceptable value, the program stops. Otherwise, control is returned to MAIN.

Next, subroutine BEGIN is called to position the data file to the desired starting point. The data files used with RMPS tend to be very large, and the program must examine each line to see if it is a record for the activity and time period being analyzed. If the user is interested in a particular time period, and he happens to know the approximate location of these data on the tape, he can specify that a number of lines in the data file be skipped, thus decreasing the program's execution time. For example, if it is known that the data of interest begin somewhere after the first 8000 lines of the file, variable ib can be set to 8000. Subroutine BEGIN would then skip the first 7999 lines without examining the data contained therein. If the location of the data is not known, ib should be set to 1, and BEGIN will not be called.

Next, the section that reads and compiles the data is entered, as subroutine READ is called. Data for the Interstate and Other State Highway system are tabulated separately, as well as together for the Total highway system. READ first reads the record type, activity number, and month on a data record and calls subroutine CKRD1 (ChecK ReaD 1) to check if these values appear reasonable.

CKRD1 will try to catch possible typographical errors, or errors in the input format for these variables. An error message is printed if the card type is not 91, 93, 91, or 9m, if the activity number is not between 200 and 299, or if the month is not an integer from 1 through 12. Control then returns to READ.

If an error had been detected, READ will go to the next data record. Also, if the data record is type 91, or 9m, READ will go to the next data record. If no errors were detected, READ calls subroutine CKRD2 (ChecK ReaD 2) to determine if the data record is for the activity and time period of interest.

For each data record, CKRD2 will determine the "status" and report it back to READ. The next action of read depends on the status. The status is reported by variable istat. If istat=0, the current record is for the correct activity and date of interest. If istat=1, the current record is either for an activity not of interest, or is for a date before the beginning of the analysis period. If istat=3, it is believed that all data for the analysis period have been read. CKRD2 first checks the activity. If the record is not for the specified activity, istat is set to 1, and control returns immediately to READ.

If the activity is correct, the date is checked. The year of the record is checked. If it is less than the year of interest, istat is set to 1 and control returns to READ. If the year is the one of interest, the month is checked. If the month is too early, istat is set to 1 and control returns to READ. If the month is too late, the next 200 records are checked to insure that all the data for the analysis period has been read. This is required because the data are entered only roughly in chronological order. A record with a date after the analysis period can occur on the data file before other records within the analysis period. If all of the next 200 records are dated later than the analysis period, istat is set to 2, and control is returned to READ. If one of the records is dated within the analysis period, istat is set to 0 before returning to READ. If the year read from a record is later than the analysis period, the next 200 records are checked to insure that all data for the analysis period have been read, as in the case when the month is too late.

It should be noted that this program was developed using data that had been sorted according to activity. In checking to insure that all data for the analysis period have been read, checking the next 200 records appeared to be adequate. However, if a different data file is used, 200 records may not be enough to insure that all data of interest have been read. This value may be changed by altering the "data look /200/" statement at the beginning of CKRD2. It should also be noted that for activities in which materials are used, 200 records means 100 crew day cards, since two records are used per crew day card.

When control returns to READ, the value of istat reported by CKRD2 is checked. If istat=1, the record is too early, so it is skipped, and the next record is read. If istat=2, it is believed that all of the data for the analysis period have been read, and control returns to MAIN. If istat=0, the record is for the correct activity and date, so the data file is backspaced one record, and a subroutine to read and tabulate all of the data on the record is called. If the record contains labor information (type 91), subroutine RDLAB (ReaD LABor) is called. If it contains material information (type 93), subroutine RDMAT (ReaD MATerial) is called.

RDLAB reads the following data on the record:

type	record type (should be 91)
act	activity number
class	highway class (INT or OSH)
unit	management unit (subdistrict number)
date	date (MMDDYY)
route	highway route type (I, US, or SR)
no	highway route number
co	county number
crew	crew size
rh	regular labor hours
ot	overtime labor hours
acc	accomplishment

The values of crew, regular and overtime hours, and accomplishment are checked to see if they are less than or equal to the maximum expected values specified by the input parameters. If any value is larger than expected, an advisory message is printed, that data record is not tabulated, and control is returned to READ. If the values are not larger than expected, class is checked. If class is neither 'int' nor 'osh', an error message is printed and control returns to READ. If the class is 'int', subroutine INTLAB (INTerstate LABor) is called. If the class is 'osh', subroutine OSHLAB (Other State Highway LABor) is called.

INTLAB tabulates Interstate labor information with two sets of variables, one for the Interstate system, and a second for the Total (int and osh) system. The following variables are used:

lacc	total number of times the activity was performed
tacc	total accomplishment
tcrew	total crew
lrh	number of times regular labor hours were reported
trh	total number of regular labor hours reported
rhacc	total accomplishment when regular labor hours were reported
lot	number of times overtime labor hours were reported
tot	total number of overtime labor hours reported
otacc	total accomplishment when overtime labor hours were reported

The same tabulations are made with a second set of variables for the Total system. These variables are named as those above with a "t" appended.

After tabulating the data, control returns to READ, and the next record is read. Subroutine OSHLAB is similar to INTLAB, with tabulations being made for the Other State

Highway and Total systems.

If the record is a material record instead of a labor record, subroutine RDMAT (ReaD MATerial) is called. RDMAT is similar to RDLAB in that it determines which highway class the record is for and calls either INTMAT (INTerstate MATerial) or OSHLAB (Other State Highway MATerial) to tabulate the data.

RDMAT reads the following information from the labor record:

type	record type (should be 93)
act	activity number
class	highway class (INT or OSH)
unit	management unit (subdistrict number)
date	date (MMDDYY)
route	highway route type (I, US, or SR)
no	highway route number
co	county number
m(j)	material code for the j-th material on the record where j=1,5
q(j)	quantity of the j-th material reported

If a class other than 'int' or 'osh' is read, an error message will be printed, and control returned to READ, which will read the next record. Otherwise, subroutine INTMAT will be called for a interstate record, and subroutine OSHMAT will be called for an Other State Highway record.

INTMAT checks each material, m(j), to see if it is one of the materials specified by the input parameters, sm(k), where k=1,nmat, and nmat is the total number of materials specified. If it is not one of the specified materials, a data check message is printed, and the next material on the record is checked. If it is one of the specified materials, the quantity is checked to see if it is greater than the maximum expected quantity, qmax(k), as specified in the input parameter file. If it is larger, a data check message is printed, and the next material on the record is checked. The following variables are tabulated from records with acceptable material codes and quantities:

lm total number of material records read (NOTE: this is tabulated for all material records before checks are made on the material codes and quantities.)

qsm# total amount of specified material #, where #=1,nmat, and nmat is the total number of materials specified in the input parameter file

lsm# total number of times specified material # is used

sm#ac total accomplishment reported when specified material # was used (NOTE: Accomplishment is recorded on labor records. Because labor records are followed immediately by the corresponding material record, this value is carried to this subroutine from READ, and is equal to the accomplishment value from the last labor record read.)

The same tabulations are made with a second set of variables for the Total (INT + OSH) system. These variables are named as those above with a "t" appended. Control returns to

READ, and the next data record is read. OSHMAT parallels INTMAT, making the same tabulations for the Other State Highway and Total systems.

When all of the records for the analysis period have been read, or the end of the data file is reached, READ returns to the MAIN program. MAIN then calls subroutine OUTKEY (OUTput KEY) to print an explanation of the program output. MAIN then enters the data analysis portion of the program. Three parallel subroutines, CALINT (CALculate INTerstate), CALOSH (CALculate Other State Highway), and CALTOT (CALculate TOTal) perform the analysis for the Interstate, Other State Highway, and Total systems, respectively. Only the systems that were specified by the user will be analyzed. The variables intout, oshout, and totout, specified as input parameters, will be equal to 0 if analysis for that system is not to be done. READ checks these, and calls the appropriate subroutines. Also, READ will not call a subroutine for a system for which there are no data records for the analysis period. Subroutine CALINT will be explained. The only differences between CALINT, CALOSH, and CALTOT are the data used.

CALINT first sets the value of variable iclass to 0 to indicate to the subroutines it calls that the highway class is INT. (Iclass is set to 1 for OSH, and 2 for TOT.) CALINT then calls subroutine CALLAB (CALculate LABor) to calculate and print labor information. Next, subroutine CALMAT

(CALculate MATerial) is called to calculate and print the material use information. Subroutine COST then calculates and prints the average cost summary, and subroutine DSTSUM (District and State SUMmary) calculates a summary of labor, material, and cost information for each district and the state as a whole. If a deviation analysis is to be done, subroutine DEVCST (DEViate CoST) is called, and finally, if any bar charts are to be printed, the appropriate subroutines are called.

Using information tabulated in subroutine INTLAB or OSHLAB, CALLAB calculates the following factors:

aacc average accomplishment = tacc / lacc

acrew average crew size = tcrew / lacc

arh average regular hours per crew day when regular hours are used = trh / lrh

oarh overall average regular hours per day = trh / lacc (NOTE: This currently is not printed in output. 11/14/83)

arhacc average accomplishment when regular are reported = rhacc / lrh

frh fraction of time regular hours were reported = lrh / lacc

prh regular hour productivity = trh / rhacc

aot average overtime hours per crew day when overtime hours are reported = tot / lot

oaot overall average overtime hours per crew day = tot / lacc (NOTE: This currently is not printed in output. 11/14/83)

arhacc average total accomplishment when overtime hours are reported = otacc / lot fot — fraction of time overtime hours were reported = lot / lacc

pot — overtime productivity = tot / otacc

pth — total labor productivity = (trh + tot) / tacc

NOTE: In making the above calculations, 0.0000000001 is added to each denominator to avoid the possibility of a floating divide by zero error.

When the quantities have been calculated, subroutine PRLAB (PRint LABor) is called to print the results, and control is returned to CALINT, which calls subroutine CALMAT (CALculate MATerial information).

CALMAT uses data tabulated in INTMAT or OSHMAT to calculate the following factors:

fsm# — fraction of time specified material # is used = lsm# / lacc, where #=1,6

asm# — average quantity of specified material # used per crew day when that material is reported = qsm# / lsm#

psm# — productivity with specified material # = qsm# / sm#ac

NOTE: In calculating the above factors, 0.0000000001 is added to each denominator to avoid the possibility of a floating divide by zero error.

CALMAT calls subroutine PRMAT (PRint MATerial factors) to print the factors just calculated, and control returns to CALINT, which then calls subroutine COST to calculate average cost factors.

COST uses information calculated in other routines to determine the average cost per accomplishment unit for each subdistrict. The cost is for labor and materials only, and is calculated as follows:

acrh — average cost for regular labor = (lacc * frh * arh * crh) / tacc

acot — average cost for overtime labor = (lacc * fot * aot * cot) / tacc

acsm# — average cost for specified material # = (lacc * fsm# * asm# * csm(#)) / tacc, where #=1,6 and csm(#)=unit cost for specified material #

NOTE: In the above calculations, 0.0000000001 is added to each denominator to avoid a possible floating divide by zero error.

aclab — average cost for labor = acrh + acot

acmat — average cost for all materials = acsm1 + acsm2 + ... + acsm6

actot — average total cost = aclab + acmat

COST then calls subroutine PRCOST (PRint COST) to print the cost figures just calculated. Then COST returns to CALINT, which calls subroutine DSTSUM.

DSTSUM calls five subroutines to calculate and print district and state summaries for labor, material, and cost. First, DSTLAB (District and STate LABor) uses subdistrict summary data calculated in other subroutines to calculate the following district and state summary information:

ldacc total number of times activity was performed in the district = sum of lacc for each district

dtacc district total accomplishment = sum of tacc for each district

dcrew total of crew sizes in a district = sum of tcrew for each district

ldrh total number of times regular labor hours were reported = sum of lrh for a district

dtrh district total regular hours used = sum of trh for a district

drhac district accomplishment when regular hours were reported = sum of rhacc for a district

ldot same a ldrh but with overtime hours

dtot same as dtrh but with overtime hours

dotac same as drhac but with overtime hours

lsacc total number of times activity was performed in the state = sum of all lacc values

tsacc state total accomplishment = sum of all tacc values

screw total of crew sizes in the state = sum of all tcrew values

lsrh total number of times regular labor hours were reported in the state = sum of all lrh values

strh state total regular hours used = sum of all trh values

srhac state accomplishment when regular hours were reported = sum of all rhacc values

lsot same as lsrh but with overtime hours

stot same as strh but with overtime hours

sotac same as srhac but with overtime hours

The following factors are calculated for each district:

adacc average district accomplishment = dtacc /ldacc

adcrw average district crew size = dcrew /ldacc

adrh average district regular hours = dtrh /ldrh

adrac average district accomplishment when regular hours are reported = drhac / ldrh

dfrh fraction of the time regular hours were reported = ldrh / ldacc

dprh district regular hour productivity = dtrh / drhac

adot average district overtime = doth / ldot

adoac average accomplishment when overtime labor is reported = dotac / ldot

dfot fraction of the time overtime labor was reported = ldot / ldacc

dpot overtime productivity = dtot / dotac

Similar factors are calculated for the state as a whole:

asacc average state accomplishment = tsacc / lsacc

ascrw average state crew size = screw / lsacc

asrh average state regular labor hours = strh / lsrh

asrac average accomplishment when regular hours are reported = srhac / lsrh

sfrh fraction of the time regular hours were reported = lsrh / lsacc

sprh state regular hour productivity = strh / srhac

asot average state overtime = soth / lsot

asoac average accomplishment when overtime labor is reported = sotac / lsot

sfot fraction of the time overtime labor was reported = lsot / lsacc spot overtime productivity = stot / sotac

NOTE: In the above calculations, 0.0000000001 is added to each denominator to avoid a possible floating divide by zero error.

Next, subroutine DSTMAT (District and STate MATerial) calculates the material summary for each district and the state as a whole. Using information calculated in other subroutines, the following variables are tabulated:

dqsm# total amount of specified material # used in a district = sum of qsm# for each district where #=1,6

ldsm# total number of times specified material # was used in the district = sum of lsm# for each district

dm#ac total district accomplishment when specified material # was used = sum of sm#ac for each district

sqsm# total state amount of specified material # used = sum of all qsm# values

lssm# total number of times specified material # was used in the state = sum of all lsm# values

sacm# total state work accomplished when specified material # was used = sum of all sm#ac values

The following factors are calculated for each district:

dfsm# fraction of time specified material # was used = ldsm# /ldacc

dasm# average quantity of specified material # used per crew day when that material was used = dqsm# / ldsm#

dpsm# productivity with specified material # = dqsm# / dm#ac

Similar calculations are made for the state as a whole:

sfsm# fraction of time specified material # was used = lssm# / lsacc

sasm# average quantity of specified material # used per crew day when that material was used = sqsm# / lssm#

spsm# productivity with specified material # = sqsm# / sacm#

NOTE: In calculating the above factors, 0.0000000001 is added to each denominator to avoid a possible floating divide by zero error.

Subroutine DSTCST (District and STate COST) uses information from other subroutines to calculate district and state cost summaries. Cost is calculated as follows: resource cost per unit of accomplishment = (number of times activity was performed * fraction of the time the resource was used * average quantity of resource when it was used * unit cost of resource) / total accomplishment.

The following factors are calculated for each district:

dacrh average cost for regular labor = (ldacc * dfrh * adrh * crh) / tdacc

dacot average cost for overtime labor = (ldacc * dfot * adot * cot) / tdacc

dacsm# average cost for specified material # = (ldacc * dfsm# * dasm# * csm(#)) / tdacc, where #=1,6

NOTE: In the above calculations, 0.0000000001 is added to each denominator to avoid a possible floating divide by zero error.

daclab average cost for labor = dacrh + dacot

dacmat average cost for all materials = dacsm1 + dacsm2 + ... + dacsm6

dactot average total cost for labor and materials = daclab + dacmat

Similar calculations are made for the state as a whole:

sacrh average cost for regular labor = (lsacc * sfrh * asrh * crh) / tsacc

sacot average cost for overtime labor = (lsacc * sfot * asot * cot) / tsacc

sacsm# average cost for specified material # = (lsacc * sfsm# * sasm# * csm(#)) / tsacc, where #=1,6

NOTE: In the above calculations, 0.0000000001 is added to each denominator to avoid a possible floating divide by zero error.

saclab average cost for labor = sacrh + sacot

sacmat average cost for all materials = sacsm1 + sacsm2 + ... + sacsm6

sactot average total cost for labor and materials = saclab + sacmat

Next, subroutine PDIST (Print DISTrict) is called to print the district labor, material, and cost summary, and subroutine PSTATE (Print STATE) is called to print the state labor, material, and cost summary.

If a deviation analysis is to be performed, CALINT calls subroutine DEVCST (DEViation by CoST). DEVCST takes the average total cost for each subdistrict, calculates the average and standard deviation, and calculates an upper and lower limit based on the average plus or minus a specified number of standard deviations. The number of standard deviations is specified by the analyst as input parameter sdev. A subdistrict is identified as being deviate if its average cost falls outside these limits. A list of the deviate units and their average cost figures is printed along with the overall average, standard deviation, and limits.

Finally, CALINT prints the bar charts that have been specified by the analyst. Up to six charts may be printed. For each chart, there is an input parameter, the value of which is zero if the chart is not to be printed or one if it is to be printed. These variables are icost, ihours, itacc, iavacc, and icrew for charts of average cost, average labor hours, total accomplishment, average accomplishment, and average crew size, respectively. The variable to specify a chart of the average quantity of one of the materials is imat, and it has an integer value of zero through six. Zero indicates that no material chart is to be printed. Another value indicates that a chart for that specified material is to be printed. For example, an imat value of three means print a bar chart of average material use per accomplishment for the third material specified in the input parameter file.

The value of each chart parameter is tested, and if it is other than zero, variable ichart is given an integer value from 0 to 5. Ichart=0 for a cost chart; ichart=1 for a labor chart; ichart=2 for a total accomplishment chart; ichart=3 for an average accomplishment chart; ichart=4 for an average crew size chart, and ichart=5 for a material quantity chart. Ichart tells the bar chart routine what type the chart is, and thus which heading to print. For each chart to be printed, CALINT calls subroutine STDEV (STandard DEViation) to calculate the average and standard deviation of the values to be plotted, and a set of limits used in the chart:

vmax	maximum value of the data being plotted
avg	average
sd	standard deviation
am3	avg - (3 * sd)
am2	avg - (2 * sd)
am1	avg - (1 * sd)
ap1	avg + (1 * sd)
ap2	avg + (2 * sd)
ap3	avg + (3 * sd)

CALINT then calls subroutine BAR to print the bar chart.

BAR first prints the heading for the bar chart. Each chart is slightly different in format, so there is a subroutine to print the top portion of each chart. These routines are CSTTOP (CoST TOP), HRSTOP (HouRS TOP), TACTOP Total ACcomplishment TOP), AACTOP (Average Accomplishment TOP), CRWTOP (CReW TOP), and MATTOP

(MATerial TOP), for cost, labor hours, total accomplishment, average accomplishment, crew size, and material use, respectively. The "top" routines print the chart page heading, chart title, and column headings for the figures listed at the right side of each chart. At the top and bottom of the chart, a line is printed indicating the position of the average and limits, am3, am2, am1, ap1, ap2, and ap3. However, the chart is scaled according to the maximum value, and some of the limits may fall outside the range of the chart. There are 16 possibilities for what set of limits falls within the chart range, each one requiring a different format. The appropriate format is determined, and noted by setting the variable wcode equal to an integer from 1 to 16. The chart line indicting the limits is then printed. The method for determining the appropriate format is explained in the comments at the beginning of BAR, and is not repeated here.

Next, BAR calls a subroutine to print the 37 bars of the chart. Along with each bar, the value the bar represents is printed on the right side of the chart alongside the values of three other factors, which vary from chart to chart. Because the charts differ slightly, different formats are required for printing the bars and factor values. BAR calls subroutine BARCST, BARHRS, BARTAC, or BARAAC to print the bars.

BARCST prints the bars for the cost chart; BARHRS prints the bars for the labor hours, crew size, and material use charts; BARTAC prints the bars for the total accomplishment chart, and BARAAC prints the bars for the average accomplishment chart. Finally, BAR repeats the line indicating the position of the limits at the bottom of the chart, and prints a legend.

At this point, CALINT returns to MAIN. If analyses for the OSH and/or TOTAL systems are to be done, CALOSH and/or CALTOT are called. CALOSH is parallel to CALINT, performing the same calculations with Other State Highway data. CALTOT is also parallel, performing the calculations with Total system data. The program then ends.

APPENDIX B

COMPUTER CODE LISTING OF PROGRAM RMPS

```
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*   Program RMPS -- Routine Maintenance Productivity Summary         **
c*                                                                    **
c*   Written by:  V. Alan Sanderson                                   **
c*                Graduate Research Assistant                         **
c*                School of Civil Engineering                         **
c*                Purdue University                                   **
c*                West Lafayette, Indiana                             **
c*                January, 1984                                       **
c*                                                                    **
c*   This program was written as part of a study of productivity of   **
c*   routine highway maintenance in the Indiana Department of High-   **
c*   ways.                                                            **
c*   The study was conducted and funded as an HPR Part II Study in    **
c*   cooperation with the Indiana Department of Highways and the      **
c*   Federal Highway Administration.                                  **
c*   Consult the user's manual for an explanation of the program and  **
c*   its use.                                                         **
c*                                                                    **
c*                                                                    **
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    MAIN                                                            **
c*                                                                    **
c*    This program will calculate various factors summarized from the **
c*    analysis of routine maintentance crew day card records.         **
c*                                                                    **
c*    Consult the user's manual for an explanation of the program     **
c*    structure and calculation procedure.                            **
c*                                                                    **
c*    This routine controls the program, switching among the          **
c*    various sections.  First input parameters are read and checked **
c*    for errors, then the data are read and tabulated, and finally   **
c*    the data are summarized, and tables and charts of various       **
c*    factors are printed.                                            **
c*                                                                    **
c*    This routine also initializes several of the program variables  **
c*    to zero.  The meanings of the variables are not explained here, **
c*    because they are explained in the various subroutines.          **
c*                                                                    **
c*                                                                    **
c***********************************************************************
c***********************************************************************
c
      integer sm,tcrewi,tcrewo,trhi,trho,toti,toto,tcrewt,trht,tott,
     *        oshout,totout
      character dsm*40,usm*20,dact*40,uact*20
c
c     The state has 37 subdistricts with numbers ranging up to 66;
c       thus, variables that are tabulated for each subdistrict are
```

```
c       dimensioned to 66.
c       "tacci(12)" is the value of variable tacci for subdistrict 1200,
c       "tacci(13)" is the value for subdistrict 1300, and so on.
      dimension sm(6),csm(6),tacci(66),tacco(66),tcrewi(66),tcrewo(66),
     *          trhi(66),trho(66),toti(66),toto(66),rhacci(66),
     *          rhacco(66),otacci(66),otacco(66),lacci(66),lacco(66),
     *          lrhi(66),lrho(66),loti(66),loto(66),qsmli(66),qsmlo(66),
     *          qsm2i(66),qsm2o(66),qsm3i(66),qsm3o(66),qsm4i(66),
     *          qsm4o(66),qsm5i(66),qsm5o(66),qsm6i(66),qsm6o(66),
     *          lsmli(66),lsmlo(66),lsm2i(66),lsm2o(66),lsm3i(66),
     *          lsm3o(66),lsm4i(66),lsm4o(66),lsm5i(66),lsm5o(66),
     *          lsm6i(66),lsm6o(66),lmi(66),lmo(66),qmax(6),
     *          smlaci(66),smlaco(66),sm2aci(66),sm2aco(66),
     *          sm3aci(66),sm3aco(66),sm4aci(66),sm4aco(66),
     *          sm5aci(66),sm5aco(66),sm6aci(66),sm6aco(66),
     *          dsm(6),usm(6)
      dimension tacct(66),tcrewt(66),trht(66),tott(66),rhacct(66),
     *          otacct(66),lacct(66),lrht(66),lott(66),qsmlt(66),
     *          qsm2t(66),qsm3t(66),qsm4t(66),qsm5t(66),qsm6t(66),
     *          lsmlt(66),lsm2t(66),lsm3t(66),lsm4t(66),lsm5t(66),
     *          lsm6t(66),lmt(66),smlact(66),sm2act(66),sm3act(66),
     *          sm4act(66),sm5act(66),sm6act(66)
c
c     Unit 1 is the file contining program control parameters
c       which tell the program what activity to analyzed
c       what the analysis period is, and what type of output
c       to provide.
c     Unit 2 is the data file.  This contains data from crew
c       day cards.
c     Unit 3 is the output file.
      open(unit=1,file='param',status='old',form='formatted')
      open(unit=2,file='data',status='old',form='formatted')
      open(unit=3,file='rpt',status='new',form='formatted')
      rewind 1
      rewind 2
      rewind 3
c
c     Initialize variables to 0.
      j1=11
      j2=16
 10   do 11 j=j1,j2
      tacci(j)=0.
      tacco(j)=0.
      tacct(j)=0.
      tcrewi(j)=0
      tcrewo(j)=0
      tcrewt(j)=0
      trhi(j)=0
      trho(j)=0
      trht(j)=0
      toti(j)=0
      toto(j)=0
      tott(j)=0
```

```
rhacci(j)=0.
rhacco(j)=0.
rhacct(j)=0.
otacci(j)=0.
otacco(j)=0.
otacct(j)=0.
lacci(j)=0
lacco(j)=0
lacct(j)=0
lrhi(j)=0
lrho(j)=0
lrht(j)=0
loti(j)=0
loto(j)=0
lott(j)=0
qsm1i(j)=0.
qsm1o(j)=0.
qsm1t(j)=0.
qsm2i(j)=0.
qsm2o(j)=0.
qsm2t(j)=0.
qsm3i(j)=0.
qsm3o(j)=0.
qsm3t(j)=0.
qsm4i(j)=0.
qsm4o(j)=0.
qsm4t(j)=0.
qsm5i(j)=0.
qsm5o(j)=0.
qsm5t(j)=0.
qsm6i(j)=0.
qsm6o(j)=0.
qsm6t(j)=0.
lsm1i(j)=0
lsm1o(j)=0
lsm1t(j)=0
lsm2i(j)=0
lsm2o(j)=0
lsm2t(j)=0
lsm3i(j)=0
lsm3o(j)=0
lsm3t(j)=0
lsm4i(j)=0
lsm4o(j)=0
lsm4t(j)=0
lsm5i(j)=0
lsm5o(j)=0
lsm5t(j)=0
lsm6i(j)=0
lsm6o(j)=0
lsm6t(j)=0
lmi(j)=0
lmo(j)=0
```

```
      lmt(j)=0
 11   continue
      if(j1.eq.11)go to 12
      if(j1.eq.21)go to 13
      if(j1.eq.31)go to 14
      if(j1.eq.41)go to 15
      if(j1.eq.51)go to 16
      if(j1.eq.61)go to 17
 12   j1=21
      j2=26
      go to 10
 13   j1=31
      j2=36
      go to 10
 14   j1=41
      j2=47
      go to 10
 15   j1=51
      j2=56
      go to 10
 16   j1=61
      j2=66
      go to 10
 17   continue
c
c     Read program control parameters
      call param(ib,iact,nmat,sm,csm,qmax,crh,cot,maxcr,maxhr,acmax,
     *           ibm,iby,iem,iey,idev,sdev,dsm,usm,dact,uact,intout,
     *           oshout,totout,icost,ihours,itacc,iavacc,icrew,imat)
c
c     Position data file to specified beginning record, ib
c       if other than 1
      if(ib.eq.1)go to 170
      call begin(ib,i)
 170  continue
c
c     Read data
c       Subroutine READ will read all the data for the given
c       activity and analysis period before returning to
c       MAIN.
      call read(iact,nmat,sm,qmax,maxcr,maxhr,acmax,ibm,iby,iem,iey,
     *          l1abi,l1abo,tacci,
     *          tacco,tcrewi,tcrewo,trhi,trho,toti,toto,rhacci,
     *          rhacco,otacci,otacco,lacci,lacco,lrhi,lrho,loti,
     *          loto,lmati,lmato,qsm1i,qsm1o,qsm2i,qsm2o,qsm3i,qsm3o,
     *          qsm4i,qsm4o,qsm5i,qsm5o,qsm6i,qsm6o,lsm1i,lsm1o,lsm2i,
     *          lsm2o,lsm3i,lsm3o,lsm4i,lsm4o,lsm5i,lsm5o,lsm6i,lsm6o,
     *          lmi,lmo,
     *          sm1aci,sm1aco,sm2aci,sm2aco,sm3aci,sm3aco,
     *          sm4aci,sm4aco,sm5aci,sm5aco,sm6aci,sm6aco,
     *          l1abt,tacct,tcrewt,trht,tott,rhacct,otacct,lacct,
     *          lrht,lott,lmatt,qsm1t,qsm2t,qsm3t,qsm4t,qsm5t,qsm6t,
     *          lsm1t,lsm2t,lsm3t,lsm4t,lsm5t,lsm6t,lmt,sm1act,
     *          sm2act,sm3act,sm4act,sm5act,sm6act)
c
c     Print a page explaining the output
      call outkey
c
c     In some cases, no work will have been done for the analysis
c       period.  If this is the case, there is no need to call the
c       routines to calculate the various factors of interest.
c       To check this, sum the number of crew days reported
c       (lacc values) and check before calling each calculation
c       routine.
c       Laccit=total number of interstate (int) crew day cards read
c       Laccot=total number of other state highway (osh) crew
c              day cards read
c       Lacctt=total number of total(int + osh) crew day cards read
      j1=10
      j2=16
 18   do 19 j=j1,j2
      laccit=laccit+lacci(j)
      laccot=laccot+lacco(j)
      lacctt=lacctt+lacct(j)
 19   continue
      if(j1.eq.10)go to 30
      if(j1.eq.20)go to 31
      if(j1.eq.30)go to 32
      if(j1.eq.40)go to 33
      if(j1.eq.50)go to 34
      if(j1.eq.60)go to 35
 30   j1=20
      j2=26
      go to 18
 31   j1=30
      j2=36
      go to 18
 32   j1=40
      j2=47
      go to 18
 33   j1=50
      j2=56
      go to 18
 34   j1=60
      j2=66
      go to 18
 35   continue
c
c     Calculate factors and print results for each subdistrict
c     If intout=0, analysis for the interstate system should
c       not be done.  Intout is one of the control parameters
c       read by PARAM.
      if(intout.eq.0)go to 20
c
c     Check to see if zero work was reported.  If so, skip
c       calculation routine and print message.
```

```
c         Laccit is the total number of crew day card records
c         included in the analysis of the interstate system.
        if(laccit.gt.0)go to 23
        write(3,900)iact,dact,ibm,iby,iem,iey
 900    format(1h1,39hROUTINE MAINTENANCE REPORT FOR ACTIVITY,1x,
      *        i3,2x,a40,5x,4hFROM,1x,i2,1h-,i2,1x,7hTHROUGH,1x,
      *        i2,1h-,i2)
        write(3,901)
 901    format(//,43hNO INSTANCES OF THIS ACTIVITY WERE REPORTED,1x,
      *       52hFOR THE INTERSTATE SYSTEM DURING THE ANALYSIS PERIOD)
        go to 20
 23     continue
c
c       Call subroutine CALINT to calculate  and print the factors for
c         the interstate system.
        call calint(iact,ibm,iby,iem,iey,nmat,sm,csm,crh,cot,llabi,
      *              tacci,tcrewi,trhi,toti,rhacci,otacci,lacci,
      *              lrhi,loti,lmati,qsm1i,qsm2i,qsm3i,qsm4i,qsm5i,
      *              qsm6i,lsm1i,lsm2i,lsm3i,lsm4i,lsm5i,lsm6i,lmi,
      *              idev,sdev,sm1aci,sm2aci,sm3aci,sm4aci,sm5aci,sm6aci,
      *              dsm,usm,dact,uact,icost,ihours,itacc,iavacc,icrew,imat)
 20     continue
c
c       Check if analysis for the other state highway system should
c         be done.  If oshout=0, it shouldn't be done.
        if(oshout.eq.0)go to 21
c
c       Check to see if there was no work reported during the
c         analysis period.  Laccot will equal zero if no work was
c         reported, and the calculation routine should be skipped.
        if(laccot.gt.0)go to 24
        write(3,900)iact,dct,ibm,iby,iem,iey
        write(3,902)
 902    format(//,51hNO INSTANCES OF THIS ACTIVITY WERE REPORTED FOR THE,
      *     1x,53hOTHER STATE HIGHWAY SYSTEM DURING THE ANALYSIS PERIOD)
        go to 21
 24     continue
c
c       Call subroutine CALOSH to calculate and print factors for
c         the other state highway system.
        call calosh(iact,ibm,iby,iem,iey,nmat,sm,csm,crh,cot,llabo,
      *              tacco,tcrewo,trho,toto,rhacco,otacco,lacco,
      *              lrho,loto,lmato,qsm1o,qsm2o,qsm3o,qsm4o,qsm5o,
      *              qsm6o,lsm1o,lsm2o,lsm3o,lsm4o,lsm5o,lsm6o,lmo,
      *              idev,sdev,sm1aco,sm2aco,sm3aco,sm4aco,sm5aco,sm6aco,
      *              dsm,usm,dact,uact,icost,ihours,itacc,iavacc,icrew,imat)
 21     continue
c
c       Check if analysis should be done for the total (int + osh)
c         system.  Totout=0 if analysis should not be done.
        if(totout.eq.0)go to 22
c
c       Check if no work was reported for the period.  If lacctt=0,
c       no work was reported, and the calculation routine should
c       be skipped.
      if(lacctt.gt.0)go to 25
      write(3,900)iact,dact,ibm,iby,iem,iey
      write(3,903)
 903  format(//,51hNO INSTANCES OF THIS ACTIVITY WERE REPORTED FOR THE,
     *    1x,51hTOTAL (INT + OSH) SYSTEM DURING THE ANALYSIS PERIOD)
      go to 22
 25   continue
c
c     Call subroutine CALTOT to calculate and print factors for
c       the total(int + osh) system.
      call caltot(iact,ibm,iby,iem,iey,nmat,sm,csm,crh,cot,llabt,
     *           tacct,tcrewt,trht,tott,rhacct,otacct,lacct,
     *           lrht,lott,lmatt,qsm1t,qsm2t,qsm3t,qsm4t,qsm5t,
     *           qsm6t,lsm1t,lsm2t,lsm3t,lsm4t,lsm5t,lsm6t,lmt,
     *           idev,sdev,sm1act,sm2act,sm3act,sm4act,sm5act,sm6act,
     *           dsm,usm,dact,uact,icost,ihours,itacc,iavacc,icrew,imat)
 22   continue
c
c     The program stops here after making all requested calculations,
c       or if no work was reported for the analysis period.
      stop
      end
      subroutine param(ib,iact,nmat,sm,csm,qmax,crh,cot,maxcr,maxhr,
     *                 acmax,ibm,iby,iem,iey,idev,sdev,
     *                 dsm,usm,dact,uact,intout,oshout,totout,
     *                 icost,ihours,itacc,iavacc,icrew,imat)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    PARAM                                                           **
c*    Subroutine PARAM reads the parameters defining what activity is **
c*    to be analyzed, its associated materials, the dates for which   **
c*    the analysis is to be conducted, and the beginning and ending   **
c*    data records of interest (can be used to skip data at beginning **
c*    of file).                                                       **
c*                                                                    **
c*    List of variables, type, definition:                            **
c*                                                                    **
c*      ib       int      beginning record (first to be read from     **
c*                                          data file)                **
c*      iact     int      IDOH activity code #                        **
c*      dact     char     Activity name (may be up to 40 characters)  **
c*      uact     char     Activity unit of measure (up to 20 characters)**
c*      nmat     int      Number of materials associated with activity **
c*                          (maximum of six)                          **
c*      sm(k)    int      IDOH code number for kth specified material **
c*                          (k ranges from 0 to nmat.  If nmat=0, no  **
c*                           sm(k) values are entered)                **
c*      csm(k)   real     Unit cost for kth specified material        **
c*      qmax(k)  real     Maximum expected quantity of kth material   **
c*                          (This value is used to check for typo-    **
c*                        graphical errors in the crew day card      **
c*                        data.  If a quantity of the kth specified  **
c*                        material is read that is greater than      **
c*                        qmax(k), that record will not be used      **
c*                        in the analysis.                           **
c*      dsm(k)  char    Description of kth specified material        **
c*                        (Up to 40 characters)                      **
c*      usm(k)  char    Unit of measure for kth specified material   **
c*                        (Up to 20 characters)                      **
c*      crh     real    Average labor wage for regular-time hour     **
c*      cot     real    Average labor wage for overtime hour         **
c*      maxcr   inte    Maximum expected crew size (Used to check for **
c*                        (for typographical errors as with qmax(k)  **
c*      maxhr   int     Maximum expected number of labor hours (Used **
c*                        to check for typographical errors.  Both   **
c*                        the number of regular and overtime hours   **
c*                        are checked with this value.)              **
c*      acmax   real    Maximum expected accomplishment (Used to     **
c*                        check for typographical errors.)           **
c*      ibm     int     First month of analysis period (January=1,   **
c*                        (February=2, etc.)                         **
c*      iby     int     Year of first month of analysis period.      **
c*                        (Only enter last two digits of year,       **
c*                         e.g., 82 for 1982.)                       **
c*      iem     int     Last month of analysis period.               **
c*      iey     int     Year of last month of analysis period.       **
c*                        (iey must be greater than or equal to iby, **
c*                         but can't be greater than iby + 1)        **
c*      idev    int     0 — deviation analysis is not to be done     **
c*                      1 — deviation analysis is to be done         **
c*      intout  int     1 — analyze interstate (int) system          **
c*                      0 — do not analyze interstate system         **
c*      oshout  int     1 — analyze other state highway (osh) system **
c*                      0 — do not analyze osh system                **
c*      totout  int     1 — analyze total (int + osh) system         **
c*                      0 — do not analyze total system              **
c*      icost   int     0 — do not print bar chart of cost           **
c*                      1 — print bar chart of average cost          **
c*      ihours  int     0 — do not print bar chart of labor hours    **
c*                          per unit of accomplishment               **
c*                      1 — print bar chart of labor hours per unit  **
c*                          of accomplishment                        **
c*      itacc   int     0 — do not print bar chart of total accom-   **
c*                          plishment                                **
c*                      1 — print bar chart of total accomplishment  **
c*      iavacc  int     0 — do not print bar chart of average accom- **
c*                          plishment                                **
c*                      1 — print bar chart of average accomplish-   **
c*                          ment                                     **
c*      icrew   int     0 — do not print bar chart of average crew   **
c*                          size                                     **
c*                      1 — print bar chart of average crew size     **
c*      imat    int     0 — do not print bar chart of average quan-  **
```

```
c*                           tity of material per accomplishment unit **
c*                      n — (where n is 1-6) print bar chart of aver-**
c*                           age quantity of specified material n per **
c*                           accomplishment unit when material n was  **
c*                           used                                     **
c*      sdev    real    The number of standard deviations to be used  **
c*                        in identifying deviate subdistricts.        **
c*                        A subdistrict will be identified as deviate **
c*                        if its average cost falls outside the range **
c*                        of the average plus or minus sdev number of **
c*                        standard deviations. (This is only entered  **
c*                        if idev=1)                                  **
c*                                                                    **
c*    Input is free format from file identified with unit 1, and      **
c*      organized as follows:                                         **
c*                                                                    **
c*      Line 1: ib                                                    **
c*      Line 2: iact nmat                                             **
c*      Line 3: dact                                                  **
c*      Line 4: uact                                                  **
c*      Line 5: sm(k) csm(k) qmax(k)  (Where k=1,nmat)                **
c*      Line 6: dsm(k)                                                **
c*      Line 7: usm(k)                                                **
c*      Lines 5-7 are repeated for each specified material giving a   **
c*        total of (3 * nmat) lines describing the materials          **
c*        If nmat=0, skip these lines.                                **
c*      Line  4+(3*nmat):  ibm iby iem iey                            **
c*      Line  5+(3*nmat):  idev intout oshout totout                  **
c*                         icost ihours itacc iavacc icrew imat       **
c*      Line  6+(3*nmat):  sdev (Skip this line if idev=0)            **
c*                                                                    **
c*                                                                    **
c**********************************************************************
c**********************************************************************
c
c
      character dsm*40, usm*20,dact*40,uact*20
      integer sm,oshout,totout
      dimension sm(6),dsm(6),usm(6),csm(6),qmax(6)
c
c     Read beginning record number, ib
      read(1,*)ib
c
c     Read activity number and number of associated materials, iact and
c       nmat
      read(1,*)iact,nmat
      read(1,900)dact
      read(1,901)uact
c
c     Read specified material code numbers
      do 11 j=1,nmat
c
c     If less than six materials were specified, set the other cost
```

```
c       values equal to zero.  This is necessary, because calculations
c       are made using all of the material cost values later in the
c       program.  That is, certain factors are calculated in loops for
c       six materials no matter how many materials are specifed.
      if(nmat.eq.6)go to 20
      do 21 k=nmat+1,6
      csm(k)=0.
 21   continue
 20   continue
c
      read(1,*)sm(j),csm(j),qmax(j)
      read(1,900)dsm(j)
 900  format(a40)
      read(1,901)usm(j)
 901  format(a20)
 11   continue
c     Read labor cost information
      read(1,*)crh,cot
c
c     Read max crew, hours, and accomplishment
      read(1,*)maxcr,maxhr,acmax
c
c     Read beginning and ending dates
      read(1,*)ibm,iby,iem,iey
c
c     Read options for deviation analysis, highway classes, and
c       bar charts
      read(1,*)idev,intout,oshout,totout,icost,ihours,itacc,iavacc,
     *         icrew,imat
      if(idev.eq.0)go to 12
      read(1,*)sdev
 12   continue
c
c     Call subroutine CKPRAM to see if parameters appear to have been
c       input correctly
      call ckpram(ib,iact,nmat,sm,csm,dsm,usm,qmax,crh,cot,
     *        maxcr,maxhr,acmax,ibm,iby,iem,iey,idev,sdev,k,dact,uact,
     *        intout,oshout,totout,icost,ihours,itacc,iavacc,icrew,imat)
      continue
c
c     If input errors were detected, k will equal something other than
c       zero.  If k is not equal to 0, stop the program.
      if(k.eq.0)go to 10
      write(3,*)'Program stopped in subroutine PARAM due to suspected pa
     *rameter input error(s)'
      stop
 10   continue
c
c     Return to MAIN
      return
      end
      subroutine ckpram(ib,iact,nmat,sm,csm,dsm,usm,qmax,crh,cot,
     *        maxcr,maxhr,acmax,ibm,iby,iem,iey,idev,sdev,k,dact,uact,
     *        intout,oshout,totout,icost,ihours,itacc,iavacc,icrew,imat)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*   CKPRAM                                                           **
c*   This subroutine will check the parameters in an effort to detect **
c*     input errors.                                                  **
c*                                                                    **
c*                                                                    **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      character dsm*40,usm*20,dact*40,uact*20
      integer sm,oshout,totout
      dimension sm(6),dsm(6),usm(6),csm(6),qmax(6)
c
c     First echo parameters
c     Print heading
      write(3,920)
 920  format(1x,26hROUTINE MAINTENANCE REPORT,//)
      write(3,900)
 900  format(1x,16hINPUT PARAMETERS)
c
c     Print activity number, description, and unit of measure
      write(3,902)iact,dact,uact
 902  format(//,1x,8hACTIVITY,3x,i3,3x,a40,3x,
     *       20hACCOMPLISHMENT UNIT:,3x,a20)
c
c     If no materials are specified, print message and skip individual
c       listing of materials.
      if(nmat.ne.0)go to 10
      write(3,941)
 941  format(/,1x,44hNO MATERIALS ARE SPECIFIED FOR THIS ACTIVITY)
      go to 16
 10   continue
c
c     Write the number, name, unit, cost, and qmax for each specified
c       material.
      write(3,921)nmat
 921  format(/,1x,3hTHE,1x,i2,1x,
     *       42hMATERIALS SPECIFIED FOR THIS ACTIVITY ARE:)
      write(3,940)
 940  format(/,6x,4hCODE,5x,11hDESCRIPTION,29x,12hUNIT COST($),
     *       2x,4hUNIT,16x,21hMAX EXPECTED QUANTITY,/,
     *       6x,4h----,5x,11(1h-),29x,12(1h-),2x,4h----,16x,21(1h-))
      do 11 j=1,nmat
      write(3,922)sm(j),dsm(j),csm(j),usm(j),qmax(j)
 922  format(6x,i4,5x,a40,3x,f7.2,4x,a20,6x,f10.2)
 11   continue
 16   continue
c     Write labor cost
      write(3,923)crh,cot
 923  format(/,1x,11hLABOR COST:,/,6x,12hRegular Hour,2x,f6.2,/,
     *      6x,13hOvertime Hour,1x,f6.2)
c
c     Write check values for crew, hours, and accomplishment
      write(3,924)maxcr,maxhr,acmax
 924  format(/,1x,25hVALUES FOR CHECKING DATA:,/,5x,8hMax Crew,8x,i7,
     *       /,5x,9hMax Hours,7x,i7,/,5x,14hMax Production,2x,f7.1)
c
c     Write beginning and ending dates
      write(3,903)ibm,iby,iem,iey
 903  format(//,1x,16hANALYSIS PERIOD:,3x,i2,1h-,i2,
     *      3x,7hthrough,3x,i2,1h-,i2)
c
c
c     Write output options
      write(3,931)
 931  format(/,1x,13hANALYSIS FOR:)
      if(intout.eq.0)go to 1
      if(intout.ne.1)go to 1
      write(3,930)
 930  format(16x,10hINTERSTATE)
 1    continue
      if(oshout.eq.0)go to 2
      if(oshout.ne.1)go to 2
      write(3,932)
 932  format(16x,19hOTHER STATE HIGHWAY)
 2    continue
      if(totout.eq.0)go to 3
      if(totout.ne.1)go to 3
      write(3,934)
 934  format(16x,24hTOTAL SYSTEM (INT + OSH))
 3    continue
c
c     Check deviation analysis option
      if(idev.eq.0)go to 12
      if(idev.ne.1)go to 13
      write(3,925)sdev
 925  format(/,1x,27hDEVIATIONS WILL BE DETECTED,1x,
     *      37hUSING COST PRODUCTIVITY BEYOND + OR -,f6.3,
     *      1x,21hSTANDARD DEVIATION(S))
      go to 13
 12   continue
      write(3,926)
 926  format(/,1x,30hNO DEVIATIONS WILL BE DETECTED)
 13   continue
c
c     Check bar chart printing options
      if((icost.ne.0).or.(itacc.ne.0).or.(iavacc.ne.0).or.(icrew.ne.0))
     *   go to 200
      write(3,950)
 950  format(/,1x,29hNO BAR CHARTS WILL BE PRINTED)
      go to 204
 200  continue
      write(3,951)
```

```
  951 format(/,1x,16hPRINT CHART FOR:)
      if(icost.eq.0)go to 201
      if(icost.ne.1)go to 201
      write(3,952)
  952 format(17x,36hAVERAGE COST PER ACCOMPLISHMENT UNIT)
  201 continue
      if(ihours.eq.0)go to 205
      if(ihours.ne.1)go to 205
      write(3,960)
  960 format(17x,35hLABOR HOURS PER ACCOMPLISHMENT UNIT)
  205 continue
      if(itacc.eq.0)go to 202
      if(itacc.ne.1)go to 202
      write(3,953)
  953 format(17x,27hTOTAL PERIOD ACCOMPLISHMENT)
  202 continue
      if(iavacc.eq.0)go to 203
      if(iavacc.ne.1)go to 203
      write(3,954)
  954 format(17x,28hAVERAGE DAILY ACCOMPLISHMENT)
  203 continue
      if(icrew.eq.0)go to 204
      if(icrew.ne.1)go to 204
      write(3,955)
  955 format(17x,17hAVERAGE CREW SIZE)
  204 continue
      if(imat.eq.0)go to 2041
      if((imat.ne.1).and.(imat.ne.2).and.(imat.ne.3).and.
     *   (imat.ne.4).and.(imat.ne.5).and.(imat.ne.6))go to 2041
      write(3,962)sm(imat)
  962 format(17x,20hQUANTITY OF MATERIAL,1x,i4,1x,
     *       23hPER ACCOMPLISHMENT UNIT)
 2041 continue
      write(3,901)ib
  901 format(//,1x,17hBeginning Record=,3x,i6)
c
c     The previous section echoed the parameters as input.
c       This section checks the parameters to see if their values
c       are reasonable.
c      Let k=number of errors detected
       k=0
c
c     Check that activity code numbers are between 200 and 299
      if((iact.gt.199).and.(iact.lt.300))go to 20
      k=k+1
      write(3,905)
  905 format(//,1x,11h***ERROR***,1x,
     * 60hSpecified activity code is less than 200 or greater than 299)
   20 continue
c
c     A maximum of six materials may be specified
      if(nmat.le.6)go to 30
      k=k+1
      write(3,906)
 906  format(//,1x,11h***ERROR***,1x,
     *       35hMore than 6 materials are specified)
 30   continue
c
c     Check that material codes are within correct range,
c       4060-4522, non-inclusive.  Range of codes determined from
c       IDOH 1982-83 Field Operations Handbookfor Foremen,
c       Division of Maintenance.
      do 40 j=1,nmat
      if((sm(j).le.4060).or.(sm(j).ge.4522))go to 50
 40   continue
      go to 60
 50   k=k+1
      write(3,907)
 907  format(//,1x,11h***ERROR***,1x,
     * 57hAll material codes are not within correct numerical range)
 60   continue
c
c     Check that beginning date is before ending date, and that
c       the months are between 1 and 12
      if((ibm.ge.1).and.(ibm.le.12))go to 70
      k=k+1
      write(3,908)
 908  format(//,1x,11h***ERROR***,1x,
     *  49hBeginning month is less than 1 or greater than 12)
 70   continue
      if((iem.ge.1).and.(iem.le.12))go to 80
      k=k+1
      write(3,909)
 909  format(//,1x,11h***ERROR***,1x,
     *  46hEnding month is less than 1 or greater than 12)
 80   continue
c
c     Check that ending year is greater than or equal to beginning
c       year, and that ending year is not later than beginning
c       year + 1.
      iy=iey-iby
      if(iy.ge.0)go to 90
      k=k+1
      write(3,910)
 910  format(//,1x,11h***ERROR***,1x,
     *       36hEnding year is before beginning year)
 90   continue
      if(iy.le.1)go to 100
      k=k+1
      write(3,911)
 911  format(//,1x,11h***ERROR***,1x,
     * 44hEnding year is later than beginning year + 1)
 100  continue
      if(iy.ne.0)go to 110
      if(iem.ge.ibm)go to 110
      k=k+1
      write(3,912)
 912  format(//,1x,11h***ERROR***,1x,
     *        36hBeginning date is before ending date)
 110  continue
c
c     Check idev
      if((idev.eq.0).or.(idev.eq.1))go to 120

      k=k+1
      write(3,914)idev
 914  format(//,1x,11h***ERROR***,1x,
     *        36hVALUE OF IDEV IS NOT CORRECT,  IDEV=,i2)
 120  continue
c
c     Check output option parameters
      if((intout.eq.0).or.(intout.eq.1))go to 130
      k=k+1
      write(3,936)intout
 936  format(/,1x,11h***ERROR***,1x,28hVALUE OF INTOUT IS INCORRECT,
     *        2x,7hINTOUT=,i4)
 130  continue
      if((oshout.eq.0).or.(oshout.eq.1))go to 140
      k=k+1
      write(3,937)oshout
 937  format(/,1x,11h***ERROR***,1x,28hVALUE OF OSHOUT IS INCORRECT,
     *        2x,7hOSHOUT=,i4)
 140  continue
      if((totout.eq.0).or.(totout.eq.1))go to 150
      k=k+1
      write(3,938)totout
 938  format(/,1x,11h***ERROR***,1x,28hVALUE OF TOTOUT IS INCORRECT,
     *        2x,7hTOTOUT=,i4)
 150  continue
c
c     Check bar chart option parameters.
      if((icost.eq.0).or.(icost.eq.1))go to 160
      k=k+1
      write(3,956)icost
 956  format(/,1x,11h***ERROR***,1x,27hVALUE OF ICOST IS INCORRECT,
     *        2x,6hICOST=,i4)
 160  continue
      if((ihours.eq.0).or.(ihours.eq.1))go to 165
      k=k+1
      write(3,961)ihours
 961  format(/,1x,11h***ERROR***,1x,28hVALUE OF IHOURS IS INCORRECT,
     *        2x,7hIHOURS=,i4)
 165  continue
      if((imat.eq.0).or.(imat.eq.1).or.(imat.eq.2).or.(imat.eq.3).or.
     *   (imat.eq.4).or.(imat.eq.5).or.(imat.eq.6))go to 167
      k=k+1
      write(3,963)imat
 963  format(/,1x,11h***ERROR***,1x,26hVALUE OF IMAT IS INCORRECT,2x,
     *         5hIMAT=,i4)
```

```
  167 continue
      if((itacc.eq.0).or.(itacc.eq.1))go to 170
      k=k+1
      write(3,957)itacc
  957 format(/,1x,11h***ERROR***,1x,27hVALUE OF ITACC IS INCORRECT,
     *       2x,6hITACC=,i4)
  170 continue
      if((iavacc.eq.0).or.(iavacc.eq.1))go to 180
      k=k+1
      write(3,958)iavacc
  958 format(/,1x,11h***ERROR***,28hVALUE OF IAVACC IS INCORRECT,
     *       2x,7hIAVACC=,i4)
  180 continue
      if((icrew.eq.0).or.(icrew.eq.1))go to 190
      k=k+1
      write(3,959)icrew
  959 format(/,1x,11h***ERROR***,1x,27hVALUE OF ICREW IS INCORRECT,
     *       2x,6hICREW=,i4)
  190 continue
c
c     Print total number of errors detected if any
      if(k.eq.0)go to 300
      write(3,913)k
  913 format(//,1x,i2,1x,36hparameter input errors were detected,)
  300 continue
c
c     Skip to new page
      write(3,915)
  915 format(1h1)
c
c     Return to subroutine PARAM
      return
      end
      subroutine begin(ib,i)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    BEGIN                                                           **
c*    This subroutine will position the data file at the desired      **
c*      beginning record.  (File has already been rewound.)           **
c*    The program examines each line of the data file to determine    **
c*      if that line contains data for the activity and time period   **
c*      of interest.  With very large data files, this process can    **
c*      a great deal of time.  However, if the approximate position   **
c*      of the data of interest is known, this subroutine can save    **
c*      some time.  For example, if it is known that the data of      **
c*      interest begins somewhere after the first 8000 lines of the   **
c*      file, the parameter ib should be set to 8000.  Then BEGIN     **
c*      will skip the first 7999 lines of the file without testing    **
c*      for the correct activity and date, saving some time.          **
c*      Line 8000 of the file will be the first one checked.          **
c*      If the location of the data within the file is not known,     **
c*      ib should be set equal to 1.                                  **
c*                                                                    **
c*                                                                    **
c***********************************************************************
c***********************************************************************
c
c     Initilize record counter, i (i=# of record to be read next)
      i=1
 10   if(i.eq.ib)go to 20
      read(2,900,end=1)a
 900  format(a1)
      i=i+1
      go to 10
c
c     If the end of the file is reached before the beginning record,
c       a message is printed.
 1    write(3,901)
 901  format(1h1,1x,26h***ABNORMAL TERMINATION***)
      write(3,*)'End of file reached before ib'
      write(3,*)'Program terminated within subroutine BEGIN'
      stop
 20   return
      end
      subroutine read(iact,nmat,sm,qmax,maxcr,maxhr,acmax,
     *                ibm,iby,iem,iey,llabi,llabo,tacci,
     *                tacco,tcrewi,tcrewo,trhi,trho,toti,toto,rhacci,
     *                rhacco,otacci,otacco,lacci,lacco,lrhi,lrho,loti,
     *                loto,lmati,lmato,qsm1i,qsm1o,qsm2i,qsm2o,qsm3i,
     *                qsm3o,qsm4i,qsm4o,qsm5i,qsm5o,qsm6i,qsm6o,lsm1i,
     *                lsm1o,lsm2i,lsm2o,lsm3i,lsm3o,lsm4i,lsm4o,lsm5i,
     *                lsm5o,lsm6i,lsm6o,lmi,lmo,
     *                sm1aci,sm1aco,sm2aci,sm2aco,sm3aci,sm3aco,
     *                sm4aci,sm4aco,sm5aci,sm5aco,sm6aci,sm6aco,
     *                llabt,tacct,tcrewt,trht,tott,rhacct,otacct,lacct,
     *               lrht,lott,lmatt,qsmlt,qsm2t,qsm3t,qsm4t,qsm5t,
     *               qsm6t,lsmlt,lsm2t,lsm3t,lsm4t,lsm5t,lsm6t,
     *               lmt,smlact,sm2act,sm3act,sm4act,sm5act,
     *               sm6act)
c*********************************************************************
c*********************************************************************
c*                                                                  **
c*    READ                                                          **
c*    This subroutine will read and tabulate crew day card records. **
c*                                                                  **
c*    First the card type, activity, and date will be read and      **
c*      and checked.  If these are for the activity and time period **
c*      under study, the entire data record will be read and values **
c*      for reasonableness.  If all checks are ok, the proper       **
c*      If all checks are ok, the data will be sent to the proper   **
c*      subroutine will be called to tabulate the data.             **
c*                                                                  **
c*    Two types of cards (or records) are of interest: labor and    **
c*      and material.  Each crew day card contains information on   **
c*      labor, material, and equipment used.  The labor and material**
c*      data are recorded on separate lines in the data file.       **
c*      A labor record is type 91, and material record is type 93.  **
c*      Type 91 and 9m records contain information entered by       **
c*      Central Office personnel to correct mistakes found in the   **
c*      crew day card records already entered.  However, the number **
c*      of 91 and 9m records are relatively small, and since the    **
c*      program checks data for reasonableness of values, the 91 and**
c*      9m records are not tabulated by the program.                **
c*                                                                  **
c*    Separate records are kept for the interstate and the other    **
c*      state highway systems, and the program tabulates these      **
c*      separately as well as tabulating a total (int + osh) system **
c*      category.                                                   **
c*                                                                  **
c*    READ first reads the card type, activity, and date and calls  **
c*      subroutine CKRD1 to check if these values appear reasonable.**
c*      This will identify typographical errors as well as possible **
c*      errors in the data format.  If the values appear reasonable,**
c*      READ calls subroutine CKRD2 to determine if the record is for **
c*      the activity and time period being analyzed.  If the record **
c*      is one of interest for the analysis, the type (labor or mat-**
c*      ierial) is determined, and the subroutine to tabulate that  **
c*      type of data is called.  RDLAB is called for labor data,    **
c*      while RDMAT is called for material data.                    **
c*                                                                  **
c*    RDLAB or RDMAT determines if the data are from the interstate **
c*      or the other state highway system and calls the subroutine  **
c*      which tabulates the data for that particular highway class. **
c*      INTLAB tabulates interstate labor; OSHLAB tabulates other   **
c*      state highway labor; INTMAT tabulates interstate material,  **
c*      and OSHMAT tabulates other state highway material.          **
c*      These subroutines also tabulate data for the total          **
c*      (int + osh) system.                                         **
```

```
c*                                                                      **
c*    Thus, labor and material data are tabulated separately for        **
c*      the interstate, other state highway, and total (int + osh)      **
c*      highway systems.                                                **
c*                                                                      **
c*    This process is repeated for each line of the data file until     **
c*      the end of the file is reached, or until it is believed that    **
c*      all of the data for the analysis period has been read.          **
c*      READ then returns to the main program, MAIN.                    **
c*                                                                      **
c*                                                                      **
c*************************************************************************
c*************************************************************************
      integer sm,tcrewi,tcrewo,trhi,trho,toti,toto,act,yr,tcrewt,trht,
     *        tott
      character type*2
      dimension sm(6),qmax(6),tacci(66),tacco(66),tcrewi(66),tcrewo(66),
     *          trhi(66),trho(66),toti(66),toto(66),rhacci(66),
     *          rhacco(66),otacci(66),otacco(66),lacci(66),lacco(66),
     *          lrhi(66),lrho(66),loti(66),loto(66),qsm1i(66),qsm1o(66),
     *          qsm2i(66),qsm2o(66),qsm3i(66),qsm3o(66),qsm4i(66),
     *          qsm4o(66),qsm5i(66),qsm5o(66),qsm6i(66),qsm6o(66),
     *          lsm1i(66),lsm1o(66),lsm2i(66),lsm2o(66),lsm3i(66),
     *          lsm3o(66),lsm4i(66),lsm4o(66),lsm5i(66),lsm5o(66),
     *          lsm6i(66),lsm6o(66),lmi(66),lmo(66),
     *          sm1aci(66),sm1aco(66),sm2aci(66),sm2aco(66),
     *          sm3aci(66),sm3aco(66),sm4aci(66),sm4aco(66),
     *          sm5aci(66),sm5aco(66),sm6aci(66),sm6aco(66),
     *          tacct(66),tcrewt(66),trht(66),tott(66),rhacct(66),
     *          otacct(66),lacct(66),lrht(66),lott(66),qsm1t(66),
     *          qsm2t(66),qsm3t(66),qsm4t(66),qsm5t(66),qsm6t(66),
     *          lsm1t(66),lsm2t(66),lsm3t(66),lsm4t(66),lsm5t(66),
     *          lsm6t(66),lmt(66),sm1act(66),sm2act(66),sm3act(66),
     *          sm4act(66),sm5act(66),sm6act(66)
c
c     irex = # of records examined
      irex=1
c
c     Read card type, act, and date
 10   read(2,900,err=1,end=2)type,act,month,yr
 900  format(a2,i3,12x,i2,2x,i2)
c
c     Check if values read appear reasonable
      call ckrd1(type,act,month,yr,irex,k)
 11   continue
c
c     If an error has been detected, k will be greater than 0.
c       If this is the case, write the entire set of data, and
c       move to the next record.
      if(k.eq.0)go to 20
      write(3,901)k,type,act,month,yr
 901  format(//,1x,9hThe above,1x,i1,1x,
     *      29herror(s) came from this data:,a2,i3,12x,i2,2x,i2)
```

```
c
c
c     Increment the counter and read the next record
      go to 12
c
c     If an error has been made in reading, write a message.
 1    write(3,*)'Error reading type, act, month, yr in READ, stmt 10'
      write(3,902)type,act,month,yr
 902  format(1x,a2,i3,12x,i2,2x,i2)
 12   irex=irex+1
      go to 10
c
c     If the end of file is reached, note this and return to MAIN
 2    write(3,*)'End of file reached in READ, stmt 10--return to main'
      return
 20   continue
c
c     The next statement will cause the program to disregard
c     type 91 and 9m records.
      if((type.eq.'91').or.(type.eq.'9m'))go to 12
c
c     If data are of correct form, check for desired values
      call ckrd2(act,month,yr,iact,ibm,iby,iem,iey,istat,irex)
c
c     The next action depends on the value of istat as determined
c       by CKRD2.
c     If istat=0, the data is for the correct activity and date
c     If istat=1, the current record should be skipped
c     If istat=2, all data within the date range is believed to have
c       been read, and calculation of factors should follow
      if(istat.eq.1)go to 12
      if(istat.eq.2)go to 30
      if(istat.eq.0)go to 40
 30   continue
c
c     All data have been read, return to MAIN
      return
 40   continue
c
c     Activity and date are correct, read entire record according
c       to the proper format for that type of record.
      backspace 2
c
c     Type 91 is a labor record
      if(type.eq.'91')go to 50
c
c     Type 93 is a material record
      if(type.eq.'93')go to 60
c
c     Increment the counter and examine the next record
      go to 12
c
c      Type 91 card, labor
```

```
c      Call subroutine to read labor
 50   call rdlab(llabi,llabo,tacci,tacco,tcrewi,tcrewo,trhi,trho,toti,
     *           toto,rhacci,rhacco,otacci,otacco,lacci,lacco,lrhi,lrho,
     *           loti,loto,maxcr,maxhr,acmax,acc,llabt,tacct,tcrewt,
     *           trht,tott,rhacct,otacct,lacct,lrht,lott)
c
c     When control is returned from RDLAB, increment counter and
c     examine the next record
      go to 12
c     Type 93 card, material
c     Call subroutine RDMAT to read material record.
 60   call rdmat(sm,qmax,lmati,lmato,qsm1i,qsm1o,qsm2i,qsm2o,qsm3i,
     *           qsm3o,qsm4i,qsm4o,qsm5i,qsm5o,qsm6i,qsm6o,lsm1i,lsm1o,
     *           lsm2i,lsm2o,lsm3i,lsm3o,lsm4i,lsm4o,lsm5i,lsm5o,
     *           lsm6i,lsm6o,lmi,lmo,acc,
     *           sm1aci,sm1aco,sm2aci,sm2aco,sm3aci,sm3aco,
     *           sm4aci,sm4aco,sm5aci,sm5aco,sm6aci,sm6aco,lmatt,
     *           qsm1t,qsm2t,qsm3t,qsm4t,qsm5t,qsm6t,lsm1t,lsm2t,
     *           lsm3t,lsm4t,lsm5t,lsm6t,lmt,sm1act,sm2act,sm3act,
     *           sm4act,sm5act,sm6act)
c
c     When control is returned from RDMAT, increment counter and
c     examine the next record
      go to 12
      end
      subroutine ckrd1(type,act,month,yr,irex,k)
c**********************************************************************
c**********************************************************************
c*                                                                  **
c*    CKRD1                                                         **
c*    This subroutine will check variables type,act,month,yr        **
c*      to try to catch typographical errors, or a format error.    **
c*                                                                  **
c*                                                                  **
c*                                                                  **
c*                                                                  **
c**********************************************************************
c**********************************************************************
      integer act,yr
      character type*2
c     k = # of errors detected
      k=0
c
c     Check type, should be 91, 93, 91, or 9m
      if((type.eq.'91').or.(type.eq.'93').or.(type.eq.'91').or.
     *  (type.eq.'9m'))go to 10
      k=k+1
      write(3,900)type,irex
 900  format(//,1x,11h***ERROR***,1x,20hUnexpected card type,5x,
     *  5hTYPE=,2x,a2,22hRecord examined, IREX=,2x,i6)
      write(3,901)
 901  format(4x,34hError detected by subroutine CKRD1)
 10   continue
```

```
c
c     Check that act is between 200 and 299
      if((act.gt.199).and.(act.lt.300))go to 20
      k=k+1
      write(3,902)act,irex
 902  format(//1x,11h***ERROR***,1x,
     *  50hRead act code is less than 200 or greater than 299,5x,
     *  4hACT=,2x,i3,5x,13hRecord, IREX=,i6)
      write(3,901)
 20   continue
c
c     Check month
      if((month.ge.1).and.(month.le.12))go to 30
      k=k+1
      write(3,903)month,irex
 903  format(//,1x,11h***ERROR***,1x,
     *  44hMonth read is less than 1 or greater than 12,5x,6hMONTH=,
     *  2x,i2,13hRecord, IREX=,i6)
      write(3,901)
 30   continue
      return
      end
      subroutine ckrd2(act,month,yr,iact,ibm,iby,iem,iey,istat,irex)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    CKRD2                                                           **
c*    This subroutine will check act and date to see if the data      **
c*      record is one of interest for the anlysis.                    **
c*                                                                    **
c*    NOTE:  It is this subroutine that determines if all of the      **
c*           data for the analysis period has been read.  Because     **
c*           the crew day card data is not entered on the file in     **
c*           order by date, the fact that a record having a date      **
c*           later than the end of the anlaysis period does not       **
c*           indicate that all of the data for the analysis period    **
c*           has already been read.  But the crew day card data is    **
c*           entered roughly in order according to date.  Thus, to    **
c*           avoid reading every record in the file, when a date      **
c*           later than the analysis period is encountered, the next  **
c*           200 records are checked.  If all 200 are later than the  **
c*           analysis period, it is assumed that all of the records   **
c*           for the period have been read.                           **
c*           If this program is used with a file where the data are   **
c*           more widespread with respect to the date, this value     **
c*           should be increased accordingly.  This can be done by    **
c*           changing the value assigned to 'look' in the data        **
c*           statement located just below this comment section.       **
c*                                                                    **
c*           It should also be noted that 'data record' in this       **
c*           context does not necessarily mean crew day card.         **
c*           For each activity in which materials are used, there     **
c*           are two data records for each crew day card, since the   **
```

```
c*          information for labor and material are on separate      **
c*          'lines' or records in the data file.  Thus, checking the **
c*          next 200 records would be equivalent to checking the     **
c*          next 100 crew day cards.  However, for activites with no **
c*          materials, there is only one record per crew day card    **
c*          (the labor record).                                      **
c*                                                                   **
c*                                                                   **
c*********************************************************************
c*********************************************************************
      integer act,yr
c
c     The meaning and use of 'look' is explained in the NOTE above.
      data look /200/
c
c     If the record is for the activity and time period being analyzed,
c       istat will be set to 0.  In this case, READ will call the
c       subroutine to tabulate the data.
c     If the record is for another activity, or is earlier than the
c       analysis period, istat will be set to 1.  In this case, READ
c       will go to the next record in the data file.
c     If it is believed that all records for the analysis time period
c       have been read, istat will be set to 2.  In this case, READ
c       will return to the main program, MAIN.
c     istat=0 if ok
c     istat=1 if wrong act or date is too early
c     istat=2 if beyond ending date
c
c     Check activity
      if(act.eq.iact)go to 10
      istat=1
      return
 10   continue
c
c     Check date
c     First determine if specified dates are in same calendar year
      if((iey-iby).eq.1)go to 100
c
c     Since iey ad iby have already been checked, in subroutine
c       CKPRAM,they should be
c       the same value if the above go to wasn't executed.
c       That is, iey should equal iby, and we won't have to deal
c       with two different calendar years in determining if the
c       record is within the analysis period.
c
c     Check if year is too early.  If so, read the next record.
      if(yr.ge.iby)go to 20
      istat=1
      return
 20   continue
c
c     Is the year one of interest
      if(yr.gt.iby)go to 30
```

```
c
c     Yr is ok, is month too early
      if(month.ge.ibm)go to 40
      istat=1
      return
 40   continue
c
c     If month is too late, check next several records to be sure
c       we have moved into a new month on the data file.
c       If the month is not too late, use the data.
      if(month.gt.iem)go to 60
      istat=0
      return
 30   continue
c
c     Read next 'look' records to make sure we are into a new
c     calendar year on the data file.
      i=1
 50   if(i.eq.(look+1))go to 51
      read(2,900,err=1,end=63)yr
 900  format(21x,i2)
      if(yr.le.iby)go to 52
 53   i=i+1
      irex=irex+1
      go to 50
 1    write(3,*)'Error reading yr in CKRD2 stmt 50, yr=',yr
      go to 53
 52   continue
c
c     If yr .le. iby, backspace and return to statement 10 of READ
c       (Because this record is not past ending year.)
      backspace 2
      istat=1
      return
 51   continue
c
c     In this condition, we believe we are past ending date specified.
c       No more data will be read.  Return to READ.
      istat=2
      return
 60   continue
c
c     This section wil check the next several records to insure
c       we have gone beyond the specified end date.
c       If a record has the correct yr and is beyond the end month,
c       check the next record.  If not, backspace and send it back
c       to READ.
      i=1
 61   if(i.eq.(look+1))go to 63
      read(2,901,err=2,end=63)month,yr
 901  format(17x,i2,2x,i2)
      if ((month.gt.iem).and.(yr.eq.iby))go to 62
      backspace 2
      istat=1
      return
 62   continue
      i=i+1
      irex=irex+1
      go to 61
 2    write(3,*)'Error reading month,yr in CKRD2 stmt 61'
      write(3,*)'month=',month,'yr=',yr
      go to 62
 63   continue
c
c     Have read past date.  Return to READ.
      istat=2
      return
 100  continue
c
c     This section will check the date when the analysis period
c       covers part of two calendar years.
c       The anlysis period must be continuous; thus we are dealing
c       with the end of one year and beginning of the followng.
c
c     Check if yr is before beginning yr.
      if(yr.ge.iby)go to 110
      istat=1
      return
 110  continue
c
c     Is yr the first calendar yr  If yes, check month.
      if(yr.ne.iby)go to 120
      if(month.ge.ibm)go to 111
c
c     If month is before ibm, read next record.
      istat=1
      return
 111  continue
c
c     In this case, yr=iby and month is .ge. ibm, so use the data.
      istat=0
      return
 120  continue
c
c     Check if yr is iey
      if(yr.gt.iey)go to 130
c
c     Yr is iey, check if past ending month.
      if(month.gt.iem)go to 140
c
c     At this point, yr=iey and month .le. iem, so use data.
      istat=0
      return
 130  continue
c
c     This section will check the next several records to
```

```
c        make sure we are beyond the ending year in the data file.
      i=1
 131  if(i.eq.(look+1))go to 133
      read(2,900,err=3,end=133)yr
      if(yr.le.iey)go to 132
 134  i=i+1
      irex=irex+1
      go to 131
 3    write(3,*)'Error reading yr in CKRD2 stmt 131, yr=',yr
      go to 134
 132  continue
      backspace 2
      istat=1
      return
 133  continue
      istat=2
      return
 140  continue
c
c     Check next several records to make sure beyond ending month
      i=1
 141  if(i.eq.(look+1))go to 143
      read(2,901,err=4,end=143)month,yr
      if((month.gt.iem).and.(yr.eq.iey))go to 142
      backspace 2
      istat=1
      return
 142  continue
      i=i+1
      irex=irex+1
      go to 141
 4    write(3,*)'Error reading month, yr in CKRD2 stmt 141'
      write(3,*)'month=',month,'yr=',yr
      go to 142
 143  continue
c
c     At end of data
      istat=2
      return
      end
      subroutine rdlab(llabi,llabo,tacci,tacco,tcrewi,tcrewo,trhi,trho,
     *                 toti,toto,rhacci,rhacco,otacci,otacco,lacci,
     *                 lacco,lrhi,lrho,loti,loto,maxcr,maxhr,acmax,acc,
     *                 llabt,tacct,tcrewt,trht,tott,rhacct,otacct,
     *                 lacct,lrht,lott)
c************************************************************************
c************************************************************************
c*                                                                    **
c*    RDLAB                                                           **
c*    This subroutine will read and tabulate data from labor records. **
c*                                                                    **
c*    First, the labor record is read, and the values checked for     **
c*      reasonableness.  If the record is for the interstate system,  **
c*      subroutine INTLAB is called to tabulate the data.  Subroutine **
c*      OSHLAB is called if the record is for the other state highway **
c*      system.  These subroutines also tabulate the data for the     **
c*      total (int + osh) system.                                     **
c*                                                                    **
c*                                                                    **
c************************************************************************
c************************************************************************
      integer tcrewi,tcrewo,trhi,trho,toti,toto,act,unit,date,co,crew,
     *        rh,ot,tcrewt,trht,tott
      character type*2, class*3
      dimension tacci(66),tacco(66),tcrewi(66),tcrewo(66),trhi(66),
     *          trho(66),toti(66),toto(66),rhacci(66),rhacco(66),
     *          otacci(66),otacco(66),lacci(66),lacco(66),lrhi(66),
     *          lrho(66),loti(66),loto(66),tacct(66),tcrewt(66),
     *          trht(66),tott(66),rhacct(66),otacct(66),lacct(66),
     *          lrht(66),lott(66)
c
c     Check data read from crew day card record for possible errors
c
c     Let k= number of possible errors detected
      k=0
      read(2,900,err=1)type,act,class,unit,date,route,no,co,crew,rh,
     *                 ot,acc
 900  format(a2,i3,2x,a3,i2,5x,i6,a2,i3,2i2,2i4,f6.1)
c
c     Check for reasonable values of crew and hours using values
c       entered by the program user.
      if(crew.le.maxcr)go to 10
      k=k+1
      write(3,901)crew
 901  format(//,1x,16h***DATA CHECK***,1x,22hLarge crew size, crew=,i2)
      write(3,902)type,act,class,unit,date,route,no,co,crew,rh,ot,acc
 902  format(1x,12hData Record:,a2,i3,2x,a3,i2,5x,i6,a2,i3,2i2,2i4,f7.1)
 10   continue
      if((rh.le.maxhr).and.(ot.le.maxhr))go to 13
      k=k+1
      write(3,903)rh,ot
 903  format(//,1x,16h***DATA CHECK***,1x,19hLarge hour value(s),5x
     *3hrh=,i4,5x,3hot=,i4)
```

```
      write(3,902)type,act,class,unit,date,route,no,co,crew,rh,ot,acc
c
c     Check for reasonable value of accomplishment
 13   if(acc.le.acmax)go to 11
      k=k+1
      write(3,905)acc
 905  format(//,1x,16h***DATA CHECK***,1x,
     *      33hQuestionable Accomplishment Value,5x,4hACC=,1x,f7.1)
      write(3,902)type,act,class,unit,date,route,no,co,crew,rh,ot,acc
      go to 11
c
c     Note if a read error occurred, and return to READ.
 1    write(3,*)'Error reading data in RDLAB stmt 900 - 1'
      write(3,902)type,act,class,unit,date,route,no,co,crew,rh,ot,acc
      return
 11   continue
c
c     If any errors were detected, return to READ.
      if(k.eq.0)go to 12
      return
 12   continue
c
c     If no errors were detected, check class, and call appropriate
c       subroutine.
      if(class.eq.'int')go to 20
      if(class.eq.'osh')go to 30
c
c     If class is neither int nor osh, write error message, and return
c       to READ.
      write(3,904)class
 904  format(//,1x,16h***DATA CHECK***,1x,
     * 33hUnusual class designation, class=,2x,a3)
      write(3,902)type,act,class,unit,date,route,no,co,crew,rh,ot,acc
      return
 20   continue
c
c     Call subroutine INTLAB to tabulate interstate data.
      call intlab(unit,crew,rh,ot,acc,llabi,tacci,tcrewi,trhi,toti,
     *           rhacci,otacci,lacci,lrhi,loti,llabt,tacct,tcrewt,
     *           trht,tott,rhacct,otacct,lacct,lrht,lott)
      return
 30   continue
c
c     Call subroutine OSHLAB to tabulate other state highway data.
      call oshlab(unit,crew,rh,ot,acc,llabo,tacco,tcrewo,trho,toto,
     *           rhacco,otacco,lacco,lrho,loto,llabt,tacct,tcrewt,
     *           trht,tott,rhacct,otacct,lacct,lrht,lott)
      continue
      return
      end
      subroutine intlab(unit,crew,rh,ot,acc,llab,tacc,tcrew,trh,tot,
     *                 rhacc,otacc,lacc,lrh,lot,llabt,tacct,tcrewt,
     *                 trht,tott,rhacct,otacct,lacct,lrht,lott)
```

```
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    INTLAB                                                          **
c*    This subroutine tabulates data from interstate labor records.   **
c*      Two sets of tabulations are made: one for the interstate      **
c*      system, and one for the total (int + osh) system.             **
c*                                                                    **
c*    The following information is tabulated by subdistrict           **
c*      (management unit):                                            **
c*                                                                    **
c*      lacc  — the total number of times the activity was performed **
c*      tacc  — total accomplishment (amount of work done)            **
c*      tcrew — total of crew sizes in persons                        **
c*      lrh   — number of times regular-time labor hours were         **
c*              used in performance of the activity                   **
c*      trh   — total number of regular-time labor hours used         **
c*      rhacc — total accomplishment when regular-time labor hours    **
c*              were used                                             **
c*      lot   — number of time overtime labor hours were used         **
c*      tot   — total number of overtime labor hours used             **
c*      otacc — total accomplishment when overtime labor hours were   **
c*              used                                                  **
c*      NOTE:  When both regular-time and overtime are reported on    **
c*             the same crew day card, there is no way to determine   **
c*             how much of the work was done on regular-time and how  **
c*             much on overtime.  Therefore, rhacc=otacc.             **
c*                                                                    **
c*      The variables listed above tabulate the information for the   **
c*        interstate system.  A second set of variables tabulates the **
c*        information for the total (int + osh) system.  These        **
c*        total system variables are named as those listed above,     **
c*        except that a 't' is appended to each variable name.        **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      integer unit,crew,rh,ot,tcrew,trh,tot,tcrewt,trht,tott
      dimension tacc(66),tcrew(66),trh(66),tot(66),rhacc(66),otacc(66),
     *          lacc(66),lrh(66),lot(66),tacct(66),tcrewt(66),trht(66),
     *          tott(66),rhacct(66),otacct(66),lacct(66),lrht(66),
     *          lott(66)
c
c     Increment total counter, llab, unit (subdistrict) counter, lacc,
c       and add accomplishment and crew size to appropriate totals.
c
c     Add to interstate totals
      llab=llab+1
      lacc(unit)=lacc(unit)+1
      tacc(unit)=tacc(unit)+acc
      tcrew(unit)=tcrew(unit)+crew
c
c     Add to total system totals
      llabt=llabt+1
      lacct(unit)=lacct(unit)+1
      tacct(unit)=tacct(unit)+acc
      tcrewt(unit)=tcrewt(unit)+crew
c
c     Check if reg hours were reported
      if(rh.eq.0)go to 10
c
c     Add reg hours, accomplishment, to appropriate totals and
c       increment counter.
c
c     Add to interstate totals
      lrh(unit)=lrh(unit)+1
      trh(unit)=trh(unit)+rh
      rhacc(unit)=rhacc(unit)+acc
c
c     Add to total system totals
      lrht(unit)=lrht(unit)+1
      trht(unit)=trht(unit)+rh
      rhacct(unit)=rhacct(unit)+acc
 10   continue
c
c     Check if ot hours were used
      if(ot.eq.0)go to 20
c
c     If ot hours were used, add to appropriate totals
c
c     Add to interstate totals
      lot(unit)=lot(unit)+1
      tot(unit)=tot(unit)+ot
      otacc(unit)=otacc(unit)+acc
c
c     Add to total system totals
      lott(unit)=lott(unit)+1
      tott(unit)=tott(unit)+ot
      otacct(unit)=otacct(unit)+acc
 20   continue
      return
      end
      subroutine oshlab(unit,crew,rh,ot,acc,llab,tacc,tcrew,trh,tot,
     *                  rhacc,otacc,lacc,lrh,lot,llabt,tacct,tcrewt,
     *                  trht,tott,rhacct,otacct,lacct,lrht,lott)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    OSHLAB                                                          **
c*    This subroutine tabulates data from other state highway labor   **
c*      records.  Two sets of tabulations are made:  one for the      **
c*      other state highway system, and one for the total (int + osh) **
c*      system.                                                       **
c*                                                                    **
c*    The following information is tabulated by subdistrict           **
c*      (management unit):                                            **
c*                                                                    **
```

```
c*      lacc  — the total number of times the activity was performed **
c*      tacc  — total accomplishment (amount of work done)           **
c*      tcrew — total of crew sizes in persons                       **
c*      lrh   — number of times regular-time labor hours were        **
c*              used in performance of the activity                  **
c*      trh   — total number of regular-time labor hours used        **
c*      rhacc — total accomplishment when regular-time labor hours   **
c*              were used                                            **
c*      lot   — number of time overtime labor hours were used        **
c*      tot   — total number of overtime labor hours used            **
c*      otacc — total accomplishment when overtime labor hours were  **
c*              used                                                 **
c*      NOTE:  When both regular-time and overtime are reported on   **
c*             the same crew day card, there is now way to determine **
c*             how much of the work was done on regular-time and how **
c*             much on overtime.  Therefore, rhacc=otacc.            **
c*                                                                   **
c*      The variables listed above tabulate the information for the  **
c*        other state highway system.  A second set of variables     **
c*        tabulates information for the total (int + osh) system.    **
c*        These total system variables are named as those listed     **
c*        above, except that a 't' is appended to each variable name. **
c*                                                                   **
c*                                                                   **
c*********************************************************************
c*********************************************************************
      integer unit,crew,rh,ot,tcrew,trh,tot,tcrewt,trht,tott
      dimension tacc(66),tcrew(66),trh(66),tot(66),rhacc(66),otacc(66),
     *          lacc(66),lrh(66),lot(66),tacct(66),tcrewt(66),trht(66),
     *          tott(66),rhacct(66),otacct(66),lacct(66),lrht(66),
     *          lott(66)
c
c     Increment total counter, llab, unit (subdistrict) counter, lacc,
c       and add accomplishment and crew size to appropriate totals.
c
c     Add to other state highway totals
      llab=llab+1
      lacc(unit)=lacc(unit)+1
      tacc(unit)=tacc(unit)+acc
      tcrew(unit)=tcrew(unit)+crew
c
c     Add to total system totals
      llabt=llabt+1
      lacct(unit)=lacct(unit)+1
      tacct(unit)=tacct(unit)+acc
      tcrewt(unit)=tcrewt(unit)+crew
c
c     Check if reg hours were reported
      if(rh.eq.0)go to 10
c
c     Add reg hours, accomplishment to appropriate totals, and
c       increment counter.
c
```

```
c     Add to other state highway totals
      lrh(unit)=lrh(unit)+1
      trh(unit)=trh(unit)+rh
      rhacc(unit)=rhacc(unit)+acc
c
c     Add to total system totals
      lrht(unit)=lrht(unit)+1
      trht(unit)=trht(unit)+rh
      rhacct(unit)=rhacct(unit)+acc
 10   continue
c
c     Check if ot hours were used
      if(ot.eq.0)go to 20
c
c     If ot hours were used, add to appropriate totals.
c
c     Add to other state highway totals
      lot(unit)=lot(unit)+1
      tot(unit)=tot(unit)+ot
      otacc(unit)=otacc(unit)+acc
c
c     Add to total system totals
      lott(unit)=lott(unit)+1
      tott(unit)=tott(unit)+ot
      otacct(unit)=otacct(unit)+acc
 20   continue
      return
      end
      subroutine rdmat(sm,qmax,lmati,lmato,qsm1i,qsm1o,qsm2i,qsm2o,
     *           qsm3i,
     *           qsm3o,qsm4i,qsm4o,qsm5i,qsm5o,qsm6i,qsm6o,lsm1i,lsm1o,
     *           lsm2i,lsm2o,lsm3i,lsm3o,lsm4i,lsm4o,lsm5i,lsm5o,
     *           lsm6i,lsm6o,lmi,lmo,acc,
     *           sm1aci,sm1aco,sm2aci,sm2aco,sm3aci,sm3aco,
     *           sm4aci,sm4aco,sm5aci,sm5aco,sm6aci,sm6aco,
     *           lmatt,qsm1t,qsm2t,qsm3t,qsm4t,qsm5t,qsm6t,lsm1t,
     *           lsm2t,lsm3t,lsm4t,lsm5t,lsm6t,lmt,sm1act,sm2act,
     *           sm3act,sm4act,sm5act,sm6act)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    RDMAT                                                           **
c*    This subroutine will read and tabulate data from material       **
c*      records.                                                      **
c*                                                                    **
c*    First, the material record is read and checked for highway      **
c*      class.  If the record is for the interstate system, sub-      **
c*      routine INTMAT is called to tabulate the data.  Subroutine    **
c*      OSHMAT is called if the record is for the other state highway **
c*      system.  These subroutines also tabulate the data for the     **
c*      total (int + osh) highway system.                             **
c*                                                                    **
c*                                                                    **
```

```
c*********************************************************************
c*********************************************************************
      integer sm,act,unit,date,co
      character type*2, class*3
      dimension m(5),q(5),sm(6),qsm1i(66),qsm1o(66),qsm2i(66),qsm2o(66),
     *          qsm3i(66),qsm3o(66),qsm4i(66),qsm4o(66),qsm5i(66),
     *          qsm5o(66),qsm6i(66),qsm6o(66),lsm1i(66),lsm1o(66),
     *          lsm2i(66),lsm2o(66),lsm3i(66),lsm3o(66),lsm4i(66),
     *          lsm4o(66),lsm5i(66),lsm5o(66),lsm6i(66),lsm6o(66),
     *          lmi(66),lmo(66),qmax(6),
     *          sm1aci(66),sm1aco(66),sm2aci(66),sm2aco(66),
     *          sm3aci(66),sm3aco(66),sm4aci(66),sm4aco(66),
     *          sm5aci(66),sm5aco(66),sm6aci(66),sm6aco(66),
     *          qsm1t(66),qsm2t(66),qsm3t(66),qsm4t(66),qsm5t(66),
     *          qsm6t(66),lsm1t(66),lsm2t(66),lsm3t(66),lsm4t(66),
     *          lsm5t(66),lsm6t(66),lmt(66),sm1act(66),sm2act(66),
     *          sm3act(66),sm4act(66),sm5act(66),sm6act(66)
c
c     Read the material record
      read(2,900,err=1)type,act,class,unit,date,route,no,co,
     *                (m(j),q(j),j=1,5)
 900  format(a2,i3,2x,a3,i2,5x,i6,a2,i3,i2,5(i4,f6.1))
c
c     Check class and call appropriate subroutine
      if(class.eq.'int')go to 10
      if(class.eq.'osh')go to 20
c
c     If an unexpected class is read, write message and return to READ
      write(3,901)class
 901  format(//,1x,16h***DATA CHECK***,1x,
     *      33hUnusual class designation, class=,2x,a3)
      write(3,902)type,act,class,unit,date,route,no,co,
     *           (m(j),q(j),j=1,5)
 902  format(1x,12hData Record:,a2,i3,2x,a3,i2,5x,i6,a2,i3,i2,
     *       5(i4,f7.1))
      write(3,903)
 903  format(1x,17hDetected by RDMAT)
      return
c
c     Write an error message if an error in reading occurred
 1    write(3,*)'Error reading data in RDMAT stmt 900'
      write(3,902)type,act,class,unit,date,route,no,co,
     *           (m(j),q(j),j=1,5)
      return
 10   continue
c
c     Call subroutine INTMAT to tabulate interstate information
      call intmat(type,act,class,unit,date,route,no,co,m,q,sm,qmax,
     *           lmati,qsm1i,qsm2i,qsm3i,qsm4i,qsm5i,qsm6i,
     *           lsm1i,lsm2i,lsm3i,lsm4i,lsm5i,lsm6i,lmi,acc,
     *           sm1aci,sm2aci,sm3aci,sm4aci,sm5aci,sm6aci,
     *           lmatt,qsm1t,qsm2t,qsm3t,qsm4t,qsm5t,qsm6t,
     *           lsm1t,lsm2t,lsm3t,lsm4t,lsm5t,lsm6t,lmt,
     *           sm1act,sm2act,sm3act,sm4act,sm5act,sm6act)
      return
   20 continue
c
c     Call subroutine OSHMAT to tabulate other state highway information
      call oshmat(type,act,class,unit,date,route,no,co,m,q,sm,qmax,
     *           lmato,qsm1o,qsm2o,qsm3o,qsm4o,qsm5o,qsm6o,
     *           lsm1o,lsm2o,lsm3o,lsm4o,lsm5o,lsm6o,lmo,acc,
     *           sm1aco,sm2aco,sm3aco,sm4aco,sm5aco,sm6aco,
     *           lmatt,qsm1t,qsm2t,qsm3t,qsm4t,qsm5t,qsm6t,
     *           lsm1t,lsm2t,lsm3t,lsm4t,lsm5t,lsm6t,lmt,
     *           sm1act,sm2act,sm3act,sm4act,sm5act,sm6act)
      return
      end
      subroutine intmat(type,act,class,unit,date,route,no,co,m,q,sm,
     *                 qmax,lmat,qsm1,qsm2,qsm3,qsm4,qsm5,qsm6,
     *                 lsm1,lsm2,lsm3,lsm4,lsm5,lsm6,lm,acc,
     *                 sm1ac,sm2ac,sm3ac,sm4ac,sm5ac,sm6ac,
     *                 lmatt,qsm1t,qsm2t,qsm3t,qsm4t,qsm5t,qsm6t,
     *                 lsm1t,lsm2t,lsm3t,lsm4t,lsm5t,lsm6t,lmt,
     *                 sm1act,sm2act,sm3act,sm4act,sm5act,sm6act)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    INTMAT                                                          **
c*    This subroutine tabulates data from interstate material rec-    **
c*      ords.  Two sets of tabulations are made: one for the inter-   **
c*      state system, and one for the total (int + osh) system.       **
c*                                                                    **
c*    The following information is tabulated by subdistrict (manage-  **
c*      ment unit):                                                   **
c*                                                                    **
c*      lm     -- total number of material records read               **
c*      qsm1   -- total amount of specified material 1 (specified     **
c*                material 1 is the first material listed in the      **
c*                input parameter file)                               **
c*      lsm1   -- total number of times specified material 1 is used  **
c*      sm1ac  -- total work accomplished when specified material 1   **
c*                was used                                            **
c*      These three variables, qsm_, lsm_, and sm_ac, are repeated    **
c*                for each specified material, 1 through 6.           **
c*                                                                    **
c*      NOTE:  The amount of work accomplished is not listed on the   **
c*             material records; it is only listed on the labor rec-  **
c*             ords.  When the crew day card data are entered on      **
c*             computer tape, the labor record is immediately follow- **
c*             ed by the corresponding material record.  Thus, to     **
c*             tabulate the sm_ac quantities, it is assumed that the  **
c*             accomplishment value previously read by subroutine     **
c*             INTLAB corresponds with the current material record.   **
c*             If the data are entered out of this seqence, the sm_ac **
c*             quantities will not be correct.                        **
c*                                                                    **
```

```
c*    The variables listed above tabulate the information for the   **
c*      interstate system.  A second set of variables tabulates the **
c*      data for the total (int + osh) system.  These total system  **
c*      variables are named as those listed above, except that a    **
c*      't' is appended to each variable name.                      **
c*                                                                  **
c*                                                                  **
c*********************************************************************
c*********************************************************************
      integer act,unit,date,co,sm
      character type*2, class*3
      dimension m(5),q(5),sm(6),qsm1(66),qsm2(66),qsm3(66),qsm4(66),
     *          qsm5(66),qsm6(66),lsm1(66),lsm2(66),lsm3(66),lsm4(66),
     *          lsm5(66),lsm6(66),lm(66),qmax(6),
     *          sm1ac(66),sm2ac(66),sm3ac(66),sm4ac(66),sm5ac(66),
     *          sm6ac(66),qsm1t(66),qsm2t(66),qsm3t(66),qsm4t(66),
     *          qsm5t(66),qsm6t(66),lsm1t(66),lsm2t(66),lsm3t(66),
     *          lsm4t(66),lsm5t(66),lsm6t(66),lmt(66),sm1act(66),
     *          sm2act(66),sm3act(66),sm4act(66),sm5act(66),sm6act(66)
c
c     Increment unit counter
      lm(unit)=lm(unit)+1
      lmt(unit)=lmt(unit)+1
c
c     Check materials.  If material is one of the specified materials,
c       add to appropriate totals.  If not, write unspecified material
c       message.
      k=1
 10   if(k.gt.5)go to 90
      if(m(k).eq.0)go to 20
      if(m(k).eq.sm(1))go to 30
      if(m(k).eq.sm(2))go to 40
      if(m(k).eq.sm(3))go to 50
      if(m(k).eq.sm(4))go to 60
      if(m(k).eq.sm(5))go to 70
      if(m(k).eq.sm(6))go to 80
c
c     At this point, must have an unexpected material. Write message.
      write(3,900)k,m(k)
 900  format(//,1x,16h***DATA CHECK***,1x,23hUnexpected material, m(,
     *       i1,2h)=,2x,i4)
      write(3,901)type,act,class,unit,date,route,no,co,
     *           (m(j),q(j),j=1,5)
 901  format(1x,12hData Record:,a2,i3,2x,a3,i2,5x,i6,a2,i3,i2,
     *       5(i4,f7.1))
      write(3,902)
 902  format(1x,18hDetected by INTMAT)
 20   k=k+1
      go to 10
c
c     Check if quantity is greater than expected maximum quantity
 30   if(q(k).gt.qmax(1))go to 31
c
```

```
c     Add to interstate totals
      qsm1(unit)=qsm1(unit)+q(k)
      lsm1(unit)=lsm1(unit)+1
      sm1ac(unit)=sm1ac(unit)+acc
c
c     Add to total system totals
      qsm1t(unit)=qsm1t(unit)+q(k)
      lsm1t(unit)=lsm1t(unit)+1
      sm1act(unit)=sm1act(unit)+acc
      go to 20
c
c     Write message if quantity is greater than expected
 31   write(3,903)sm(1),q(k)
 903  format(/,1x,16h***DATA CHECK***,2x,
     *       23hLARGE MATERIAL QUANTITY,2x,4hMAT=,2x,i4,2x,
     *       5hQUAN=,2x,f7.1)
      write(3,901)type,act,class,unit,date,route,no,co,
     *            (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 40   if(q(k).gt.qmax(2))go to 41
c
c     Add to interstate totals
      qsm2(unit)=qsm2(unit)+q(k)
      lsm2(unit)=lsm2(unit)+1
      sm2ac(unit)=sm2ac(unit)+acc
c
c     Add to total system totals
      qsm2t(unit)=qsm2t(unit)+q(k)
      lsm2t(unit)=lsm2t(unit)+1
      sm2act(unit)=sm2act(unit)+acc
      go to 20
 41   write(3,903)sm(2),q(k)
      write(3,901)type,act,class,unit,date,route,no,co,
     *            (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 50   if(q(k).gt.qmax(3))go to 51
c
c     Add to interstate totals
      qsm3(unit)=qsm3(unit)+q(k)
      lsm3(unit)=lsm3(unit)+1
      sm3ac(unit)=sm3ac(unit)+acc
c
c     Add to total system totals
      qsm3t(unit)=qsm3t(unit)+q(k)
      lsm3t(unit)=lsm3t(unit)+1
      sm3act(unit)=sm3act(unit)+acc
      go to 20
 51   write(3,903)sm(3),q(k)
      write(3,901)type,act,class,unit,date,route,no,co,
     *            (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 60   if(q(k).gt.qmax(4))go to 61
c
c     Add to interstate totals
      qsm4(unit)=qsm4(unit)+q(k)
      lsm4(unit)=lsm4(unit)+1
      sm4ac(unit)=sm4ac(unit)+acc
c
c     Add to total system totals
      qsm4t(unit)=qsm4t(unit)+q(k)
      lsm4t(unit)=lsm4t(unit)+1
      sm4act(unit)=sm4act(unit)+acc
      go to 20
 61   write(3,903)sm(4),q(k)
      write(3,901)type,act,class,unit,date,route,no,co,
     *            (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 70   if(q(k).gt.qmax(5))go to 71
c
c     Add to interstate totals
      qsm5(unit)=qsm5(unit)+q(k)
      lsm5(unit)=lsm5(unit)+1
      sm5ac(unit)=sm5ac(unit)+acc
c
c     Add to total system totals
      qsm5t(unit)=qsm5t(unit)+q(k)
      lsm5t(unit)=lsm5t(unit)+1
      sm5act(unit)=sm5act(unit)+acc
      go to 20
 71   write(3,903)sm(5),q(k)
      write(3,901)type,act,class,unit,date,route,no,co,
     *            (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 80   if(q(k).gt.qmax(6))go to 81
c
c     Add to interstate totals
      qsm6(unit)=qsm6(unit)+q(k)
      lsm6(unit)=lsm6(unit)+1
      sm6ac(unit)=sm6ac(unit)+acc
c
c     Add to total system totals
      qsm6t(unit)=qsm6t(unit)+q(k)
      lsm6t(unit)=lsm6t(unit)+1
      sm6act(unit)=sm6act(unit)+acc
      go to 20
 81   write(3,903)sm(6),q(k)
      write(3,901)type,act,class,unit,date,route,no,co,
     *            (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 90   continue
```

```
      return
      end
      subroutine oshmat(type,act,class,unit,date,route,no,co,m,q,sm,
     *                  qmax,lmat,qsm1,qsm2,qsm3,qsm4,qsm5,qsm6,
     *                  lsm1,lsm2,lsm3,lsm4,lsm5,lsm6,lm,acc,
     *                  sm1ac,sm2ac,sm3ac,sm4ac,sm5ac,sm6ac,
     *                  lmatt,qsm1t,qsm2t,qsm3t,qsm4t,qsm5t,qsm6t,
     *                  lsm1t,lsm2t,lsm3t,lsm4t,lsm5t,lsm6t,lmt,
     *                  sm1act,sm2act,sm3act,sm4act,sm5act,sm6act)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    OSHMAT                                                          **
c*                                                                    **
c*    This sbroutine tabulates data from other state highway material **
c*      records.  Two sets of tabulations are made: one for the other **
c*      state highway system, and one for the total (int + osh) sys-  **
c*      tem.                                                          **
c*                                                                    **
c*   The following information is tabulated by subdistrict (manage-   **
c*     ment unit):                                                    **
c*                                                                    **
c*     lm    -- total number of material records read                 **
c*     qsm1  -- total amount of specified material 1 used (specified  **
c*              material 1 is the first material listed in the input  **
c*              parameter file)                                       **
c*     lsm1  -- total number of times specified material 1 was used   **
c*     sm1ac -- total work accomplished when specified material 1 was **
c*              used                                                  **
c*     These three variables, qsm_, lsm_, and am_ac, are repeated for **
c*              for each specified material, 1 through 6.             **
c*                                                                    **
c*     NOTE:  The amount of work accomplished is not listed on the    **
c*            material records; it is only listed on the labor rec-   **
c*            ords.  When the crew day card data are entered on com-  **
c*            puter tape, the labor record is immediately followed by **
c*            corresponding material record.  Thus, to tabulate the   **
c*            sm_ac quantities, it is assumed that the accomplishment **
c*            value previously read by subroutine INTLAB corresponds  **
c*            with the current material record.  If the data are not  **
c*            entered in this sequence, the sm_ac quantities will not **
c*            be correct.                                             **
c*                                                                    **
c*     The variables listed above tabulate the information for the    **
c*       other state highway system.  A second set of variables tab-  **
c*       ulates the data for the total (int + osh) system.  These     **
c*       total system variables are named as those listed above,      **
c*       except that a 't' is appended to each variable name.         **
c*                                                                    **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      integer act,unit,date,co,sm
      character type*2, class*3
      dimension m(5),q(5),sm(6),qsm1(66),qsm2(66),qsm3(66),qsm4(66),
     *          qsm5(66),qsm6(66),lsm1(66),lsm2(66),lsm3(66),lsm4(66),
     *          lsm5(66),lsm6(66),lm(66),qmax(6),
     *          sm1ac(66),sm2ac(66),sm3ac(66),sm4ac(66),sm5ac(66),
     *          sm6ac(66),qsm1t(66),qsm2t(66),qsm3t(66),qsm4t(66),
     *          qsm5t(66),qsm6t(66),lsm1t(66),lsm2t(66),lsm3t(66),
     *          lsm4t(66),lsm5t(66),lsm6t(66),sm1act(66),sm2act(66),
     *          sm3act(66),sm4act(66),sm5act(66),sm6act(66),lmt(66)
c
c     Increment unit counter
      lm(unit)=lm(unit)+1
      lmt(unit)=lmt(unit)+1
c
c     Check materials.  If material is one of the specified materials,
c       add to appropriate totals.  If not, write unspecified material
c       message.
c     Up to five materials may be listed on a material record.
      k=1
 10   if(k.gt.5)go to 90
      if(m(k).eq.0)go to 20
      if(m(k).eq.sm(1))go to 30
      if(m(k).eq.sm(2))go to 40
      if(m(k).eq.sm(3))go to 50
      if(m(k).eq.sm(4))go to 60
      if(m(k).eq.sm(5))go to 70
      if(m(k).eq.sm(6))go to 80
c
c     At this point, must have an unexpected material. Write message.
      write(3,900)k,m(k)
 900  format(//,1x,16h***DATA CHECK***,1x,23hUnexpected material, m(,
     *       i1,2h)=,2x,i4)
      write(3,901)type,act,class,unit,date,route,no,co,
     *            (m(j),q(j),j=1,5)
 901  format(1x,12hData Record:,a2,i3,2x,a3,i2,5x,i6,a2,i3,i2,
     *       5(i4,f7.1))
      write(3,902)
 902  format(1x,18hDetected by OSHMAT)
 20   k=k+1
      go to 10
c
c     Check if quantity is greater than expected maximum quantity
 30   if(q(k).gt.qmax(1))go to 31
c
c     Add to other state highway totals
      qsm1(unit)=qsm1(unit)+q(k)
      lsm1(unit)=lsm1(unit)+1
      sm1ac(unit)=sm1ac(unit)+acc
c
c     Add to total system totals
      qsm1t(unit)=qsm1t(unit)+q(k)
      lsm1t(unit)=lsm1t(unit)+1
      sm1act(unit)=sm1act(unit)+acc
      go to 20
c
```

```
c     Write message if quantity is greater than expected
 31   write(3,903)sm(1),q(k)
 903  format(/,1x,16h***DATA CHECK***,2x,
     *       23hLARGE MATERIAL QUANTITY,2x,4hMAT=,2x,i4,2x,
     *       5hQUAN=,2x,f7.1)
      write(3,901)type,act,class,unit,date,route,no,co,
     *            (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 40   if(q(k).gt.qmax(2))go to 41
c
c     Add to other state highway totals
      qsm2(unit)=qsm2(unit)+q(k)
      lsm2(unit)=lsm2(unit)+1
      sm2ac(unit)=sm2ac(unit)+acc
c
c     Add to total system totals
      qsm2t(unit)=qsm2t(unit)+q(k)
      lsm2t(unit)=lsm2t(unit)+1
      sm2act(unit)=sm2act(unit)+acc
      go to 20
 41   write(3,903)sm(2),q(k)
      write(3,901)type,act,class,unit,date,route,no,co,
     *            (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 50   if(q(k).gt.qmax(3))go to 51
c
c     Add to other state highway totals
      qsm3(unit)=qsm3(unit)+q(k)
      lsm3(unit)=lsm3(unit)+1
      sm3ac(unit)=sm3ac(unit)+acc
c
c     Add to total system totals
      qsm3t(unit)=qsm3t(unit)+q(k)
      lsm3t(unit)=lsm3t(unit)+1
      sm3act(unit)=sm3act(unit)+acc
      go to 20
 51   write(3,903)sm(3),q(k)
      write(3,901)type,act,class,unit,date,route,no,co,
     *            (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 60   if(q(k).gt.qmax(4))go to 61
c
c     Add to other state highway totals
      qsm4(unit)=qsm4(unit)+q(k)
      lsm4(unit)=lsm4(unit)+1
      sm4ac(unit)=sm4ac(unit)+acc
c
c     Add to total system totals
      qsm4t(unit)=qsm4t(unit)+q(k)
      lsm4t(unit)=lsm4t(unit)+1
      sm4act(unit)=sm4act(unit)+acc
      go to 20
 61   write(3,903)sm(4),q(k)
      write(3,901)type,act,class,unit,date,route,no,co,
     *           (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 70   if(q(k).gt.qmax(5))go to 71
c
c     Add to other state highway totals
      qsm5(unit)=qsm5(unit)+q(k)
      lsm5(unit)=lsm5(unit)+1
      sm5ac(unit)=sm5ac(unit)+acc
c
c     Add to total system totals
      qsm5t(unit)=qsm5t(unit)+q(k)
      lsm5t(unit)=lsm5t(unit)+1
      sm5act(unit)=sm5act(unit)+acc
      go to 20
 71   write(3,903)sm(5),q(k)
      write(3,901)type,act,class,unit,date,route,no,co,
     *           (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 80   if(q(k).gt.qmax(6))go to 81
c
c     Add to other state highway
      qsm6(unit)=qsm6(unit)+q(k)
      lsm6(unit)=lsm6(unit)+1
      sm6ac(unit)=sm6ac(unit)+acc
c
c     Add to total system totals
      qsm6t(unit)=qsm6t(unit)+q(k)
      lsm6t(unit)=lsm6t(unit)+1
      sm6act(unit)=sm6act(unit)+acc
      go to 20

 81   write(3,903)sm(6),q(k)
      write(3,901)type,act,class,unit,date,route,no,co,
     *           (m(j),q(j),j=1,5)
      write(3,902)
      go to 20
 90   continue
      return
      end
      subroutine outkey
c*************************************************************************
c*************************************************************************
c*                                                                     **
c*    OUTKEY                                                           **
c*                                                                     **
c*    This subroutine will print explanations helpful in interpreting **
c*      the output.                                                    **
```

```
c*                                                                      **
c*                                                                      **
c***********************************************************************
c***********************************************************************
      write(3,900)
 900  format(1h1,26hROUTINE MAINTENANCE REPORT,10x,
     *       10hOUTPUT KEY)
      write(3,901)
 901  format(//,1x,'Depending on the maintenance activity being analyzed
     * and the output options selected, up to twelve pages of results',
     * /,1x,'per highway class are produced.  These include summaries by
     * subdistrict of labor use, material use, and average cost, a',
     * /,1x,
     * 'summary by district, a statewide summary, and a list of subdistr
     *icts whose cost falls outside a specified range.',/,1x,
     * 'Up to six bar charts may be printed, providing a graphical di
     *splay of analysis results.')
      write(3,902)
 902  format(/,1x,44hThe list below will help in interpreting the,1x,
     *       7houtput.,/)
      write(3,903)
 903  format(1x,34hCOMMON TO MOST OF THE OUTPUT PAGES,/,1x,34(1h-))
      write(3,904)
 904  format(1x,43hUNIT                         -- management unit ,
     * 51h(subdistrict) number; even "1000" numbers refer to ,
     * 19hdistrict-wide crews,/,28x,
     * 40he.g., 2000 refers to crew for district 2)
      write(3,905)
 905  format(1x,27hCREW DAYS                 -- ,
     * 52hthe number of crew day card records included in the ,
     *  8hanalysis)
      write(3,906)
 906  format(/,1x,17hLABOR INFORMATION,/,1x,17(1h-))
      write(3,907)
 907  format(1x,27hACCOMPLISHMENT, TOTAL   -- ,
     *  54htotal amount of work done, measured in accomplishment ,
     *  26hunits for a given activity)
      write(3,908)
 908  format(1x,27hACCOMPLISHMENT, AVERAGE -- ,
     *  36haverage accomplishment per crew day ,
     *  34h(TOTAL ACCOMPLISHMENT / CREW DAYS))
      write(3,909)
 909  format(1x,27hAVG CREW                  -- ,
     *  28haverage crew size in persons)
      write(3,910)
 910  format(1x,27hTOT RH DAYS               -- ,
     *  48htotal number of crew day cards on which regular ,
     *  25hlabor hours were reported)
      write(3,911)
 911  format(1x,27hRH/ACC                    -- ,
     *  55haverage number of regular labor hours used per unit of ,
     *  37haccomplishment for the given activity,/,28x,
     *  51h(TOT RH / total accomplishment from crew day cards ,
    *  37hon which regular hours were reported))
     write(3,912)
912  format(1x,27hAVGRH                   -- ,
    *  51haverage number of regular labor hours reported per ,
    *  31hcrew day (TOT RH / TOT RH DAYS))
     write(3,913)
913  format(1x,27hTOT RH                  -- ,
    *  53htotal number of regular labor hours reported for the ,
    *  15hanalysis period)
     write(3,914)
914  format(1x,27hRH/TOT DAYS             -- ,
    *  51hfraction of the time regular labor hours were used ,
    *  25h(TOT RH DAYS / CREW DAYS))
     write(3,915)
915  format(1x,27hTOT OT DAYS             -- ,
    *  49htotal number of crew day cards on which overtime ,
    *  25hlabor hours were reported)
     write(3,916)
916  format(1x,27hOT/ACC                  -- ,
    *  56haverage number of overtime labor hours used per unit of ,
    *  37haccomplishment for the given activity,/,28x,
    *  51h(TOT OT / total accomplishment from crew day cards ,
    *  38hon which overtime hours were reported))
     write(3,917)
917  format(1x,27hAVGOT                   -- ,
    *  52haverage number of overtime labor hours reported per ,
    *  31hcrew day (TOT OT / TOT OT DAYS))
     write(3,918)
918  format(1x,27hTOT OT                  -- ,
    *  54htotal number of overtime labor hours reported for the ,
    *  15hanalysis period)
     write(3,919)
919  format(1x,27hOT/TOT DAYS             -- ,
    *  52hfraction of the time overtime labor hours were used ,
    *  25h(TOT OT DAYS / CREW DAYS))
     write(3,920)
920  format(/,1x,20hMATERIAL INFORMATION,/,1x,20(1h-))
     write(3,921)
921  format(1x,27hMAT XXXX                -- ,
    *  44hmaterial, where xxxx is a four-digit number ,
    *  48hcorresponding to an IDOH Maintenance Management ,/,28x,
    *  20hSystem Material Code)
     write(3,922)
922  format(1x,27hFRAC                    -- ,
    *  43hfraction of the time the material was used ,
    *  49h(number of crew cay cards on which material xxxx ,/,28x,
    *  25hwas reported / CREW DAYS))
     write(3,923)
923  format(1x,27hAVGQNT                  -- ,
    *  50haverage quantity of material used (total quantity ,
    *  42hof material xxxx reported / number of crew,/,28x,
    *  46hday cards on which material xxxx was reported))
     write(3,924)
```

```
924  format(1x,27hQNT/AC                     -- ,
   *  56haverage quantity of material per unit of accomplishment ,
   *  40h(total quantity of material xxxx / total,/,28x,
   *  20haccomplishment from ,
   *  51hcrew day cards on which material xxxx was reported))
     write(3,925)
925  format(/,1x,16hCOST INFORMATION,/,1x,16(1h-))
     write(3,926)
926  format(1x,27hTOTAL ACCOMP               -- ,
   *  20htotal accomplishment)
     write(3,927)
927  format(1x,27hTOT COST                   -- ,
   *  53haverage total cost per accomplishment unit for labor ,
   *  13hand materials)
     write(3,928)
928  format(1x,27hLAB COST                   -- ,
   *  42haverage labor cost per accomplishment unit)
     write(3,929)
929  format(1x,27hMAT COST                   -- ,
   *  44haverage material cost per accomplishmet unit)
     write(3,930)
930  format(1x,27hCOSTS BY MATERIAL          -- ,
   *  54haverage cost per accomplishment unit for each material)
     return
     end
     subroutine calint(iact,ibm,iby,iem,iey,nmat,sm,csm,crh,cot,llabi,
    *                  tacci,tcrewi,trhi,toti,rhacci,
    *                  otacci,lacci,lrhi,loti,lmati,qsm1i,qsm2i,qsm3i,
    *                  qsm4i,qsm5i,qsm6i,lsm1i,lsm2i,lsm3i,lsm4i,lsm5i,
    *                  lsm6i,lmi,idev,sdev,sm1aci,sm2aci,sm3aci,
    *                  sm4aci,sm5aci,sm6aci,dsm,usm,dact,uact,
    *                  icost,ihours,itacc,iavacc,icrew,imat)
c*********************************************************************
c*********************************************************************
c*                                                                  **
c*   CALINT                                                         **
c*                                                                  **
c*   This subroutine will call various subroutines to perform the   **
c*     desired calculations with the interstate system data.        **
c*                                                                  **
c*   First, subroutine CALLAB is called to calculate and print a    **
c*     summary of labor use.  Next, subroutine CALMAT calculates and **
c*     prints material use information.  Then subroutine COST is     **
c*     called to calculate and print average cost for performing the **
c*     activity.  A summary of labor, materials, and cost by dis-    **
c*     trict and for the state as a whole are calculated by subrou-  **
c*     tine DSTSUM.                                                  **
c*   If an analysis is to be done to identify subdistricts whose    **
c*     cost falls outside a specified number of standard deviations  **
c*     about the mean, subroutine DEVCST will be called.             **
c*                                                                  **
c*   Finally, if any bar charts are to be printed, the appropriate  **
c*     subroutines will be called to print them.                    **
```

```
c*                                                                     **
c*                                                                     **
c***********************************************************************
c***********************************************************************
      integer sm,tcrewi,trhi,toti,dunit
      character dsm*40,usm*20,dact*40,uact*20
      dimension sm(6),csm(6),tacci(66),tcrewi(66),
     *          trhi(66),toti(66),rhacci(66),
     *          otacci(66),lacci(66),lrhi(66),loti(66),qsmli(66),
     *          qsm2i(66),qsm3i(66),qsm4i(66),qsm5i(66),qsm6i(66),
     *          lsmli(66),lsm2i(66),lsm3i(66),lsm4i(66),lsm5i(66),
     *          lsm6i(66),lmi(66),arhi(66),aoti(66),frhi(66),
     *          foti(66),fsmli(66),fsm2i(66),fsm3i(66),fsm4i(66),
     *          fsm5i(66),fsm6i(66),asmli(66),asm2i(66),asm3i(66),
     *          asm4i(66),asm5i(66),asm6i(66),smlaci(66),sm2aci(66),
     *          sm3aci(66),sm4aci(66),sm5aci(66),sm6aci(66),prhi(66),
     *          poti(66),pthi(66),psmli(66),psm2i(66),psm3i(66),
     *          psm4i(66),psm5i(66),psm6i(66),dunit(20),dsm(6),usm(6),
     *          actoti(66),aacci(66),acrewi(66)
c
c     Set variable to tell summary and bar chart routines what
c       the highway class is
c       iclass=0 for interstate
        iclass=0
c
c     Call subroutine to calculate and print labor info
      call callab(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *            llabi,tacci,tcrewi,trhi,toti,rhacci,otacci,
     *            lacci,lrhi,loti,arhi,frhi,aoti,foti,prhi,poti,pthi,
     *            aacci,acrewi)
      continue
c
c     Calculate interstate material data
c
c     If no materials have been specified, skip the material routines.
      if(nmat.eq.0)go to 1
c
c     Call subroutine to calc and print material info
      call calmat(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *            sm,lmati,qsmli,qsm2i,qsm3i,qsm4i,qsm5i,qsm6i,
     *            lsmli,lsm2i,lsm3i,lsm4i,lsm5i,lsm6i,lacci,
     *            fsmli,fsm2i,fsm3i,fsm4i,fsm5i,fsm6i,asmli,asm2i,
     *            asm3i,asm4i,asm5i,asm6i,
     *            smlaci,sm2aci,sm3aci,sm4aci,sm5aci,sm6aci,
     *            psmli,psm2i,psm3i,psm4i,psm5i,psm6i)
 1    continue
c
c     Call subroutine to calculate and print interstate cost data
      call cost(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *          sm,csm,crh,cot,lacci,tacci,arhi,aoti,frhi,foti,
     *          fsmli,fsm2i,fsm3i,fsm4i,fsm5i,fsm6i,asmli,asm2i,
     *          asm3i,asm4i,asm5i,asm6i,actoti)
c
```

```
c     Call subroutine to calculate and print district and state labor
c       summaries
      call dstsum(iact,dact,uact,ibm,iby,iem,iey,iclass,tacci,tcrewi,
     *            trhi,toti,rhacci,atacci,lacci,lrhi,loti,arhi,frhi,
     *            aoti,foti,sm,qsm1i,qsm2i,qsm3i,qsm4i,qsm5i,qsm6i,
     *            lsm1i,lsm2i,lsm3i,lsm4i,lsm5i,lsm6i,sm1aci,sm2aci,
     *            sm3aci,sm4aci,sm5aci,sm6aci,csm,crh,cot)
c
c
c     Check if deviation analysis is to be done.
      if(idev.eq.0)go to 700
      go to 600
c
c
c     Find deviations based on cost per unit accomplisment
 600  continue
c
      call devcst(iact,dact,ibm,iby,iem,iey,iclass,
     *            idev,sdev,actoti,nunit,dunit,avg,uact)
 700  continue
c
c     Print desired bar charts
c
c     Chart for average cost
      if(icost.eq.0)go to 710
      ichart=0
      call stdev(actoti,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actoti,
     *         tacci,aacci,lacci,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 710  continue
c
c     Chart for labor hours
      if(ihours.eq.0)go to 750
      ichart=1
      call stdev(pthi,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actoti,
     *         tacci,aacci,pthi,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 750  continue
c
c     Chart for total accomplishment
      if(itacc.eq.0)go to 720
      ichart=2
      call stdev(tacci,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actoti,
     *         tacci,aacci,lacci,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 720  continue
c
c     Chart for average accomplishment
      if(iavacc.eq.0)go to 730
      ichart=3
      call stdev(aacci,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actoti,
     *         tacci,aacci,lacci,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 730  continue
c
c     Chart for average crew size
      if(icrew.eq.0)go to 740
      ichart=4
      call stdev(acrewi,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actoti,
     *         tacci,aacci,acrewi,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 740  continue
c
c     Chart for material productivity
      if(imat.eq.0)go to 760
      ichart=5
      if(imat.ne.1)go to 761
      call stdev(psm1i,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actoti,tacci,aacci,psm1i,vmax,avg,sd,am3,am2,
     *         am1,ap1,ap2,ap3)
      go to 760
 761  continue
      if(imat.ne.2)go to 762
      call stdev(psm2i,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actoti,tacci,aacci,psm2i,vmax,avg,sd,am3,am2,
     *         am1,ap1,ap2,ap3)
      go to 760
 762  continue
      if(imat.ne.3)go to 763
      call stdev(psm3i,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actoti,tacci,aacci,psm3i,vmax,avg,sd,am3,am2,
     *         am1,ap1,ap2,ap3)
      go to 760
 763  continue
      if(imat.ne.4)go to 764
      call stdev(psm4i,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actoti,tacci,aacci,psm4i,vmax,avg,sd,am3,am2,
     *         am1,ap1,ap2,ap3)
      go to 760
 764  continue
      if(imat.ne.5)go to 765
      call stdev(psm5i,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actoti,tacci,aacci,psm5i,vmax,avg,sd,am3,am2,
     *         am1,ap1,ap2,ap3)
      go to 760
 765  continue
```

```
      if(imat.ne.6)go to 760
      call stdev(psm6i,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actoti,tacci,aacci,psm6i,vmax,avg,sd,am3,am2,
     *         aml,apl,ap2,ap3)
760   continue
      return
      end
      subroutine calosh(iact,ibm,iby,iem,iey,nmat,sm,csm,crh,cot,llabo,
     *                  tacco,tcrewo,trho,toto,rhacco,
     *                  otacco,lacco,lrho,loto,lmato,qsmlo,qsm2o,qsm3o,
     *                  qsm4o,qsm5o,qsm6o,lsmlo,lsm2o,lsm3o,lsm4o,lsm5o,
     *                  lsm6o,lmo,idev,sdev,smlaco,sm2aco,sm3aco,
     *                  sm4aco,sm5aco,sm6aco,dsm,usm,dact,uact,
     *                  icost,ihours,itacc,iavacc,icrew,imat)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    CALOSH                                                          **
c*                                                                    **
c*    This subroutine will call various subroutines to perform the    **
c*      desired calculations with the other state highway system      **
c*      data.  It is parallel to subroutine CALINT, performing the    **
c*      same functions with other state highway data that CALINT does **
c*      with interstate data.                                         **
c*                                                                    **
c*    First, subroutine CALLAB is called to calculate and print a     **
c*      summary of labor use.  Next, subroutine CALMAT calculates and **
c*      prints material use information.  Then subroutine COST is     **
c*      called to calculate and print average cost for performing the **
c*      activity.  A summary of labor, materials, and cost by dis-    **
c*      trict and for the state as a whole are calculated by subrou-  **
c*      tine DSTSUM.                                                  **
c*    If an analysis is to be done to identify subdistricts whose     **
c*      cost falls outside a specified number of standard deviations  **
c*      about the mean, subroutine DEVCST will be called.             **
c*                                                                    **
c*    Finally, if any bar charts are to be printed, the appropriate   **
c*      subroutines will be called to print them.                     **
c*                                                                    **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      integer sm,tcrewo,trho,toto,dunit
      character dsm*40,usm*20,dact*40,uact*20
      dimension sm(6),csm(6),tacco(66),tcrewo(66),trho(66),toto(66),
     *          rhacco(66),otacco(66),lacco(66),lrho(66),loto(66),
     *          qsmlo(66),qsm2o(66),qsm3o(66),qsm4o(66),qsm5o(66),
     *          qsm6o(66),lsmlo(66),lsm2o(66),lsm3o(66),lsm4o(66),
     *          lsm5o(66),lsm6o(66),lmo(66),arho(66),aoto(66),frho(66),
     *          foto(66),fsmlo(66),fsm2o(66),fsm3o(66),fsm4o(66),
     *          fsm5o(66),fsm6o(66),asmlo(66),asm2o(66),asm3o(66),
     *          asm4o(66),asm5o(66),asm6o(66),smlaco(66),sm2aco(66),
     *          sm3aco(66),sm4aco(66),sm5aco(66),sm6aco(66),prho(66),
     *       poto(66),ptho(66),psmlo(66),psm2o(66),psm3o(66),psm4o(66),
     *          psm5o(66),psm6o(66),dunit(20),dsm(6),usm(6),actoto(66),
     *          aacco(66),acrewo(66)
c
c     Set variable to tell summary and bar chart routines highway class
c       iclass=1 for other state highway
      iclass=1
```

```
c
c     Call subroutine to calculate and print labor info
      call callab(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *            llabo,tacco,tcrewo,trho,toto,rhacco,otacco,
     *            lacco,lrho,loto,arho,frho,aoto,foto,prho,poto,ptho,
     *            aacco,acrewo)
c
c     Calculate material data
c
c     If no materials are specified, skip material routines.
      if(nmat.eq.0)go to 1
c
c     Call subroutine to calculate and print material info
      call calmat(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *            sm,lmato,qsm1o,qsm2o,qsm3o,qsm4o,qsm5o,qsm6o,
     *            lsm1o,lsm2o,lsm3o,lsm4o,lsm5o,lsm6o,lacco,
     *            fsm1o,fsm2o,fsm3o,fsm4o,fsm5o,fsm6o,asm1o,asm2o,
     *            asm3o,asm4o,asm5o,asm6o,
     *            sm1aco,sm2aco,sm3aco,sm4aco,sm5aco,sm6aco,
     *            psm1o,psm2o,psm3o,psm4o,psm5o,psm6o)
 1    continue
c
c     Call subroutine to calculate and print cost data
      call cost(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *          sm,csm,crh,cot,lacco,tacco,arho,aoto,frho,foto,
     *          fsm1o,fsm2o,fsm3o,fsm4o,fsm5o,fsm6o,asm1o,asm2o,
     *          asm3o,asm4o,asm5o,asm6o,actoto)
c
c     Calculate and print district and state labor totals
      call dstsum(iact,dact,uact,ibm,iby,iem,iey,iclass,tacco,tcrewo,
     *     trho,toto,rhacco,otacco,lacco,lrho,loto,arho,frho,aoto,
     *     foto,sm,qsm1o,qsm2o,qsm3o,qsm4o,qsm5o,qsm6o,lsm1o,
     *     lsm2o,lsm3o,lsm4o,lsm5o,lsm6o,sm1aco,sm2aco,sm3aco,
     *     sm4aco,sm5aco,sm6aco,csm,crh,cot)
c
c
c     Check if deviation analysis is to be done.
      if(idev.eq.0)go to 700
      go to 600
c
c
c     Find deviations based on cost per unit accomplisment
 600  continue
c
      call devcst(iact,dact,ibm,iby,iem,iey,iclass,
     *            idev,sdev,actoto,nunit,dunit,avg,uact)
 700  continue
c
c     Print desired bar charts
c
c     Check for average cost chart
      if(icost.eq.0)go to 710
      ichart=0
      call stdev(actoto,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actoto,
     *         tacco,aacco,lacco,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 710  continue
c
c     Check for labor hours chart
      if(ihours.eq.0)go to 750
      ichart=1
      call stdev(ptho,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actoto,
     *         tacco,aacco,ptho,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 750  continue
c
c     Check for total accomplishment chart
      if(itacc.eq.0)go to 720
      ichart=2
      call stdev(tacco,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actoto,
     *         tacco,aacco,lacco,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 720  continue
c
c     Check for average accomplishment chart
      if(iavacc.eq.0)go to 730
      ichart=3
      call stdev(aacco,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actoto,
     *         tacco,aacco,lacco,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 730  continue
c
c     Check for average crew size chart
      if(icrew.eq.0)go to 740
      ichart=4
      call stdev(acrewo,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actoto,
     *         tacco,aacco,acrewo,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 740  continue
c
c     Chart for material productivity
      if(imat.eq.0)go to 760
      ichart=5
      if(imat.ne.1)go to 761
      call stdev(psm1o,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actoto,tacco,aacco,psm1o,vmax,avg,sd,am3,am2,
     *         am1,ap1,ap2,ap3)
      go to 760
 761  continue
      if(imat.ne.2)go to 762
      call stdev(psm2o,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *        usm(imat),actoto,tacco,aacco,psm2o,vmax,avg,sd,am3,am2,
     *        am1,ap1,ap2,ap3)
      go to 760
 762  continue
      if(imat.ne.3)go to 763
      call stdev(psm3o,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *        usm(imat),actoto,tacco,aacco,psm3o,vmax,avg,sd,am3,am2,
     *        am1,ap1,ap2,ap3)
      go to 760
 763  continue
      if(imat.ne.4)go to 764
      call stdev(psm4o,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *        usm(imat),actoto,tacco,aacco,psm4o,vmax,avg,sd,am3,am2,
     *        am1,ap1,ap2,ap3)
      go to 760
 764  continue
      if(imat.ne.5)go to 765
      call stdev(psm5o,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *        usm(imat),actoto,tacco,aacco,psm5o,vmax,avg,sd,am3,am2,
     *        am1,ap1,ap2,ap3)
      go to 760
 765  continue
      if(imat.ne.6)go to 760
      call stdev(psm6o,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *        usm(imat),actoto,tacco,aacco,psm6o,vmax,avg,sd,am3,am2,
     *        am1,ap1,ap2,ap3)
 760  continue
      return
      end
      subroutine caltot(iact,ibm,iby,iem,iey,nmat,sm,csm,crh,cot,llabt,
     *                  tacct,tcrewt,trht,tott,rhacct,
     *                  otacct,lacct,lrht,lott,lmatt,qsm1t,qsm2t,qsm3t,
     *                  qsm4t,qsm5t,qsm6t,lsm1t,lsm2t,lsm3t,lsm4t,lsm5t,
     *                  lsm6t,lmt,idev,sdev,sm1act,sm2act,sm3act,
     *                  sm4act,sm5act,sm6act,dsm,usm,dact,
     *                  uact,icost,ihours,itacc,iavacc,icrew,imat)
c**********************************************************************
c**********************************************************************
c*                                                                  **
c*   CALTOT                                                         **
c*                                                                  **
c*   This subroutine will call various subroutines to perform the   **
c*     desired calculations with the total (int + osh) highway sys- **
c*     tem.  It is parallel to CALINT and CALOSH, performing the    **
c*     calculations with the total system data.                     **
c*                                                                  **
c*   First, subroutine CALLAB is called to calculate and print a    **
```

```
c*      summary of labor use.  Next, subroutine CALMAT calculates and **
c*      prints material use information.  Then subroutine COST is     **
c*      called to calculate and print average cost for performing the **
c*      activity.  A summary of labor, materials, and cost by dis-    **
c*      trict and for the state as a whole are calculated by subrou-  **
c*      tine DSTSUM.                                                  **
c*    If an analysis is to be done to identify subdistricts whose     **
c*      cost falls outside a specified number of standard deviations  **
c*      about the mean, subroutine DEVCST will be called.             **
c*                                                                    **
c*    Finally, if any bar charts are to be printed, the appropriate   **
c*      subroutines will be called to print them.                     **
c*                                                                    **
c*                                                                    **
c*********************************************************************
c*********************************************************************
      integer sm,tcrewt,trht,tott,dunit
      character dsm*40,usm*20,dact*40,uact*20
      dimension sm(6),csm(6),tacct(66),tcrewt(66),
     *          trht(66),tott(66),rhacct(66),otacct(66),lacct(66),
     *          lrht(66),lott(66),qsmlt(66),qsm2t(66),qsm3t(66),
     *          qsm4t(66),qsm5t(66),qsm6t(66),lsmlt(66),lsm2t(66),
     *          lsm3t(66),lsm4t(66),lsm5t(66),lsm6t(66),lmt(66),
     *          arht(66),aott(66),frht(66),fott(66),fsmlt(66),fsm2t(66),
     *          fsm3t(66),fsm4t(66),fsm5t(66),fsm6t(66),asmlt(66),
     *          asm2t(66),asm3t(66),asm4t(66),asm5t(66),asm6t(66),
     *          smlact(66),sm2act(66),sm3act(66),sm4act(66),sm5act(66),
     *          sm6act(66),prht(66),pott(66),ptht(66),psmlt(66),
     *          psm2t(66),psm3t(66),psm4t(66),psm5t(66),psm6t(66),
     *          dunit(20),dsm(6),usm(6),actott(66),aacct(66),acrewt(66)
c
c     Set variable to tell summary and bar chart routines highway class
c       iclass=2 for total
      iclass=2
c
c     Call subroutine to calculate and print labor info
      call callab(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *            llabt,tacct,tcrewt,trht,tott,rhacct,otacct,
     *            lacct,lrht,lott,arht,frht,aott,fott,prht,pott,ptht,
     *            aacct,acrewt)
c
c     Calculate material data
c
c     If no materials have been specified, skip material routines.
      if(nmat.eq.0)go to 1
c
c     Call subroutine to calculate and print material info
      call calmat(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *            sm,lmatt,qsmlt,qsm2t,qsm3t,qsm4t,qsm5t,qsm6t,
     *            lsmlt,lsm2t,lsm3t,lsm4t,lsm5t,lsm6t,lacct,
     *            fsmlt,fsm2t,fsm3t,fsm4t,fsm5t,fsm6t,asmlt,asm2t,
     *            asm3t,asm4t,asm5t,asm6t,
     *            smlact,sm2act,sm3act,sm4act,sm5act,sm6act,
     *           psm1t,psm2t,psm3t,psm4t,psm5t,psm6t)
 1    continue
c
c     Call subroutine to calculate and print cost data
      call cost(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *         sm,csm,crh,cot,lacct,tacct,arht,aott,frht,fott,
     *         fsm1t,fsm2t,fsm3t,fsm4t,fsm5t,fsm6t,asm1t,asm2t,
     *         asm3t,asm4t,asm5t,asm6t,actott)
c
c     Calculate and print district and state labor totals
      call dstsum(iact,dact,uact,ibm,iby,iem,iey,iclass,tacct,tcrewt,
     *     trht,tott,rhacct,otacct,lacct,lrht,lott,arht,frht,aott,
     *     fott,sm,qsm1t,qsm2t,qsm3t,qsm4t,qsm5t,qsm6t,lsm1t,lsm2t,
     *     lsm3t,lsm4t,lsm5t,lsm6t,sm1act,sm2act,sm3act,sm4act,sm5act,
     *     sm6act,csm,crh,cot)
c
c
c     Check if deviation analysis is to be done.
      if(idev.eq.0)go to 700
      go to 600
c
c
c     Find deviations based on cost per unit accomplisment
 600  continue
c
      call devcst(iact,dact,ibm,iby,iem,iey,iclass,
     *           idev,sdev,actott,nunit,dunit,avg,uact)
 700  continue
c
c     Print desired bar charts
c
c     Check for average cost chart
      if(icost.eq.0)go to 710
      ichart=0
      call stdev(actott,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *        sm(imat),usm(imat),actott,
     *        tacct,aacct,lacct,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 710  continue
c
c     Check for labor hours chart
      if(ihours.eq.0)go to 750
      ichart=1
      call stdev(ptht,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *        sm(imat),usm(imat),actott,
     *        tacct,aacc,ptht,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
 750  continue
c
c     Check for total accomplishment chart
      if(itacc.eq.0)go to 720
      ichart=2
      call stdev(tacct,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actott,
     *         tacct,aacct,lacct,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
 720  continue
c
c     Check for average accomplishment chart
      if(iavacc.eq.0)go to 730
      ichart=3
      call stdev(aacct,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actott,
     *         tacct,aacct,lacct,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
 730  continue
c
c     Check for averge crew size chart
      if(icrew.eq.0)go to 740
      ichart=4
      call stdev(acrewt,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *         sm(imat),usm(imat),actott,
     *         tacct,aacct,acrewt,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
 740  continue
c
c     Chart for material productivity
      if(imat.eq.0)go to 760
      ichart=5
      if(imat.ne.l)go to 761
      call stdev(psmlt,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actott,tacct,aacct,psmlt,vmax,avg,sd,am3,am2,
     *         aml,apl,ap2,ap3)
      go to 760
 761  continue
      if(imat.ne.2)go to 762
      call stdev(psm2t,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actott,tacct,aacct,psm2t,vmax,avg,sd,am3,am2,
     *         aml,apl,ap2,ap3)
      go to 760
 762  continue
      if(imat.ne.3)go to 763
      call stdev(psm3t,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actott,tacct,aacct,psm3t,vmax,avg,sd,am3,am2,
     *         aml,apl,ap2,ap3)
      go to 760
 763  continue
      if(imat.ne.4)go to 764
      call stdev(psm4t,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actott,tacct,aacct,psm4t,vmax,avg,sd,am3,am2,
     *         aml,apl,ap2,ap3)
      go to 760
```

```
764   continue
      if(imat.ne.5)go to 765
      call stdev(psm5t,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actott,tacct,aacct,psm5t,vmax,avg,sd,am3,am2,
     *         am1,ap1,ap2,ap3)
      go to 760
765   continue
      if(imat.ne.6)go to 760
      call stdev(psm6t,vmax,avg,sd,am3,am2,am1,ap1,ap2,ap3)
      call bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,sm(imat),
     *         usm(imat),actott,tacct,aacct,psm6t,vmax,avg,sd,am3,am2,
     *         am1,ap1,ap2,ap3)
760   continue
      return
      end
      subroutine callab(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *                 llab,tacc,tcrew,trh,tot,rhacc,otacc,lacc,
     *                 lrh,lot,arh,frh,aot,fot,prh,pot,pth,aacc,acrew)
c***********************************************************************
c***********************************************************************
c*                                                                   **
c*    CALLAB                                                         **
c*                                                                   **
c*    This subroutine calculates various factors from the labor     **
c*      information tabulated by INTLAB and OSHLAB.  This subroutine **
c*      is called in turn to calculate these factors for the inter-  **
c*      state, the other state highway, and the total highway system. **
c*                                                                   **
c*    The following information is provided by INTLAB and OSHLAB:   **
c*                                                                   **
c*      lacc  -- the total number of times the activity was performed **
c*      tacc  -- total accomplishment (amount of work done)          **
c*      tcrew -- total of crew sizes in persons                      **
c*      lrh   -- number of times regular-time labor hours were       **
c*               used in performance of the activity                 **
c*      trh   -- total number of regular-time labor hours used       **
c*      rhacc -- total accomplishment when regular-time labor hours  **
c*               were used                                           **
c*      lot   -- number of time overtime labor hours were used       **
c*      tot   -- total number of overtime labor hours used           **
c*      otacc -- total accomplishment when overtime labor hours were **
c*               used                                                **
c*                                                                   **
c*    Using the above information, the following factors are       **
c*      calculated:                                                  **
c*                                                                   **
c*      aacc   -- average accomplishment = tacc / lacc               **
c*      acrew  -- average crew size = tcrew / lacc                   **
c*      arh    -- average regular hours per crew day = trh / lrh     **
c*      oarh   -- overall average regular hours = trh / lacc         **
c*                (This currently is not printed in output. 11/14/83) **
c*      arhacc -- average regular hour accomplishment = rhacc / lrh  **
```

```
c*      frh    -- fraction of time regular-time hours were used          **
c*                = lrh / lacc                                           **
c*      prh    — regular hour productivity = trh / rhacc                 **
c*      aot    — average overtime hours per crew day = tot / lot         **
c*      oaot   — overall average overtime hours = tot / lacc             **
c*                (This currently is not printed in output. 11/14/83) **
c*      arhacc — average overtime hour accomplishment = otacc / lot  **
c*      fot    — fraction of time overtime hours were used               **
c*                = lot / lacc                                           **
c*      pot    —  overtime productivity = tot / otacc                    **
c*      pth    —  total labor productivity = (trh + tot) / tacc          **
c*                                                                       **
c*      NOTE:  In making the above calculations, 0.0000000001 is add- **
c*             ed to each denominator to avoid a floating divide by   **
c*             zero error.                                               **
c*                                                                       **
c*                                                                       **
c************************************************************************
c************************************************************************
      integer tcrew,trh,tot
      character dact*40,uact*20
      dimension tacc(66),tcrew(66),trh(66),tot(66),rhacc(66),otacc(66),
     *          lacc(66),lrh(66),lot(66),aacc(66),acrew(66),arh(66),
     *          oarh(66),arhacc(66),frh(66),aot(66),oaot(66),
     *          aotacc(66),fot(66),prh(66),pot(66),pth(66)
c
c     Calculate the quantities
c
c     Calculate for district 1 subs
      j1=10
      j2=16
 10   do 11 j=j1,j2
      aacc(j)=tacc(j) / (float(lacc(j)) + .0000000001)
      acrew(j)=float(tcrew(j)) / (float(lacc(j)) + .0000000001)
      arh(j)=float(trh(j)) / (float(lrh(j)) + .0000000001)
      oarh(j)=float(trh(j)) / (float(lacc(j)) + .0000000001)
      arhacc(j)=rhacc(j) / (float(lrh(j)) + .0000000001)
      frh(j)=float(lrh(j)) / (float(lacc(j)) + .0000000001)
      aot(j)=float(tot(j)) / (float(lot(j)) + .0000000001)
      oaot(j)=float(tot(j)) / (float(lacc(j)) + .0000000001)
      aotacc(j)=otacc(j) / (float(lot(j)) + .0000000001)
      fot(j)=float(lot(j)) / (float(lacc(j)) + .0000000001)
      prh(j)=float(trh(j)) / (rhacc(j) + .0000000001)
      pot(j)=float(tot(j)) / (otacc(j) + .0000000001)
      pth(j)=float(tot(j) + trh(j)) / (tacc(j) + .0000000001)
 11   continue
c
c     Calculate quantities for each district's subdistricts, and
c       district-wide crew
      if(j1.eq.10)go to 20
      if(j1.eq.20)go to 30
      if(j1.eq.30)go to 40
      if(j1.eq.40)go to 50
      if(j1.eq.50)go to 60
      if(j1.eq.60)go to 70
 20   j1=20
      j2=26
      go to 10
 30   j1=30
      j2=36
      go to 10
 40   j1=40
      j2=47
      go to 10
 50   j1=50
      j2=56
      go to 10
 60   j1=60
      j2=66
      go to 10
 70   continue
c
c     All quantities have been calculated for each subdistrict.
c     Print the results
      call prlab(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *           lacc,tacc,aacc,acrew,lrh,arh,oarh,arhacc,frh,
     *           lot,aot,oaot,aotacc,fot,prh,pot,trh,tot)
      continue
      return
      end
      subroutine prlab(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *                 lacc,tacc,aacc,acrew,lrh,arh,oarh,arhacc,frh,
     *                 lot,aot,oaot,aotacc,fot,prh,pot,trh,tot)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    PRLAB                                                           **
c*                                                                    **
c*    This subroutine prints the labor factors just calculated by     **
c*      subroutine CALLAB.                                            **
c*                                                                    **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      integer trh,tot
      character dact*40,uact*20
      dimension lacc(66),tacc(66),aacc(66),acrew(66),lrh(66),arh(66),
     *          oarh(66),arhacc(66),frh(66),lot(66),aot(66),oaot(66),
     *          aotacc(66),fot(66),prh(66),pot(66),trh(66),tot(66)
c
c     Print title
      write(3,904)iact,dact,ibm,iby,iem,iey,uact
 904  format(1h1,39hROUTINE MAINTENANCE REPORT FOR ACTIVITY,1x,
     *  i3,2x,a40,5x,4hFROM,1x,i2,1h-,i2,1x,7hTHROUGH,1x,i2,1h-,i2,
     *  /,21x,20hACCOMPLISHMENT UNIT:,5x,a20)
c
```

```
c     Check to see if printing interstate, other state highway, or
c       total system information and print appropriate title
      if(iclass.ne.0)go to 100
      write(3,905)
 905  format(/,1x,28hINTERSTATE LABOR INFORMATION)
      go to 120
 100  continue
      if(iclass.ne.1)go to 110
      write(3,906)
 906  format(/,1x,37hOTHER STATE HIGHWAY LABOR INFORMATION)
      go to 120
 110  continue
      write(3,907)
 907  format(/,1x,35hTOTAL (INT + OSH) LABOR INFORMATION)
 120  continue
c     Print the column headings
      write(3,900)
 900  format(///,9x,1x,4hCREW,3x,14hACCOMPLISHMENT,3x,3hAVG,4x,
     *      6hTOT RH,26x,6hRH/TOT,2x,6hTOT OT,26x,6hOT/TOT)
      write(3,901)
 901  format(2x,4hUNIT,4x,4hDAYS,3x,5hTOTAL,2x,7hAVERAGE,3x,
     *       4hCREW,4x,4hDAYS,3x,6hRH/ACC,2x,6hAVG RH,2x,6hTOT RH,
     *       3x,4hDAYS,4x,4hDAYS,3x,6hOT/ACC,2x,6hAVG OT,2x,
     *       6hTOT OT,3x,4hDAYS)
      write(3,902)
 902  format(2x,4h----,3x,6h------,13(2x,6h------),//)
c
c     Write the data for each subdistrict
      j1=10
      j2=16
 10   do 11 j=j1,j2
      write(3,903)j,lacc(j),tacc(j),aacc(j),acrew(j),lrh(j),prh(j),
     *            arh(j),trh(j),frh(j),lot(j),pot(j),aot(j),
     *            tot(j),fot(j)
 903  format(2x,i2,2h00,3x,i6,1x,f8.1,1x,f6.1,2x,f6.2,2x,i6,
     *       2(2x,f6.2),2x,i6,2x,f6.3,2x,i6,2(2x,f6.2),2x,i6,
     *       2x,f5.3)
 11   continue
      if(j1.eq.10)go to 20
      if(j1.eq.20)go to 30
      if(j1.eq.30)go to 40
      if(j1.eq.40)go to 50
      if(j1.eq.50)go to 60
      if(j1.eq.60)go to 70
 20   j1=20
      j2=26
      go to 10
 30   j1=30
      j2=36
      go to 10
 40   j1=40
      j2=47
      go to 10
```

```
50    j1=50
      j2=56
      go to 10
60    j1=60
      j2=66
      go to 10
70    continue
      return
      end
      subroutine calmat(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *                 sm,lmat,qsm1,qsm2,qsm3,qsm4,qsm5,qsm6,lsm1,
     *                 lsm2,lsm3,lsm4,lsm5,lsm6,lacc,
     *                 fsm1,fsm2,fsm3,fsm4,fsm5,fsm6,asm1,asm2,asm3,
     *                 asm4,asm5,asm6,
     *                 sm1ac,sm2ac,sm3ac,sm4ac,sm5ac,sm6ac,
     *                 psm1,psm2,psm3,psm4,psm5,psm6)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    CALMAT                                                          **
c*                                                                    **
c*    This subroutine calculates various factors from the material    **
c*      information tabulated by INTMAT and OSHMAT.  This subroutine  **
c*      is called in turn to calculate these factors for the inter-   **
c*      state, the other state highway, and the total highway system. **
c*                                                                    **
c*    The following information is provided by INTMAT and OSHMAT:     **
c*                                                                    **
c*     lacc   -- total number of times activity was performed         **
c*     qsm1  -- total amount of specified material 1 used (specified  **
c*              material 1 is the first material listed in the input  **
c*              parameter file)                                       **
c*     lsm1  -- total number of times specified material 1 was used   **
c*     sm1ac -- total work accomplished when specified material 1 was **
c*              used                                                  **
c*     These three variables, qsm_, lsm_, and am_ac, are repeated for **
c*              for each specified material, 1 through 6.             **
c*                                                                    **
c*   Using this information, the following factors are calculated:    **
c*                                                                    **
c*     fsm1  -- fraction of time specified material 1 is used         **
c*              = lsm1 / lacc                                         **
c*     asm1  -- average quantity of specified material 1 used per     **
c*              crew day when that material is used = qsm1 / lsm1    **
c*     psm1  -- productivity with specified material 1 = qsm1 / sm1ac **
c*     These three factors, fsm_, asm_, and psm_, are repeated for    **
c*              each specified material, 1 through 6.                 **
c*                                                                    **
c*     NOTE:  In calculating the above factors, 0.0000000001 is add-  **
c*            to each denominator to avoid a floating divide by zero  **
c*            error.                                                  **
c*                                                                    **
c*                                                                    **
```

```
c***********************************************************************
c***********************************************************************
      integer sm
      character dact*40,uact*20
      dimension sm(6),qsm1(66),qsm2(66),qsm3(66),qsm4(66),qsm5(66),
     *          qsm6(66),lsm1(66),lsm2(66),lsm3(66),lsm4(66),lsm5(66),
     *          lsm6(66),lacc(66),
     *          fsm1(66),fsm2(66),fsm3(66),fsm4(66),fsm5(66),fsm6(66),
     *          asm1(66),asm2(66),asm3(66),asm4(66),asm5(66),asm6(66),
     *          sm1ac(66),sm2ac(66),sm3ac(66),sm4ac(66),sm5ac(66),
     *          sm6ac(66),psm1(66),psm2(66),psm3(66),psm4(66),
     *          psm5(66),psm6(66)
      j1=10
      j2=16
 10   do 11 j=j1,j2
      fsm1(j)=float(lsm1(j)) / (float(lacc(j)) + .0000000001)
      fsm2(j)=float(lsm2(j)) / (float(lacc(j)) + .0000000001)
      fsm3(j)=float(lsm3(j)) / (float(lacc(j)) + .0000000001)
      fsm4(j)=float(lsm4(j)) / (float(lacc(j)) + .0000000001)
      fsm5(j)=float(lsm5(j)) / (float(lacc(j)) + .0000000001)
      fsm6(j)=float(lsm6(j)) / (float(lacc(j)) + .0000000001)
      asm1(j)=qsm1(j) / (float(lsm1(j)) + .0000000001)
      asm2(j)=qsm2(j) / (float(lsm2(j)) + .0000000001)
      asm3(j)=qsm3(j) / (float(lsm3(j)) + .0000000001)
      asm4(j)=qsm4(j) / (float(lsm4(j)) + .0000000001)
      asm5(j)=qsm5(j) / (float(lsm5(j)) + .0000000001)
      asm6(j)=qsm6(j) / (float(lsm6(j)) + .0000000001)
      psm1(j)=qsm1(j) / (sm1ac(j) + .0000000001)
      psm2(j)=qsm2(j) / (sm2ac(j) + .0000000001)
      psm3(j)=qsm3(j) / (sm3ac(j) + .0000000001)
      psm4(j)=qsm4(j) / (sm4ac(j) + .0000000001)
      psm5(j)=qsm5(j) / (sm5ac(j) + .0000000001)
      psm6(j)=qsm6(j) / (sm6ac(j) + .0000000001)
 11   continue
      if(j1.eq.10)go to 20
      if(j1.eq.20)go to 30
      if(j1.eq.30)go to 40
      if(j1.eq.40)go to 50
      if(j1.eq.50)go to 60
      if(j1.eq.60)go to 70
 20   j1=20
      j2=26
      go to 10
 30   j1=30
      j2=36
      go to 10
 40   j1=40
      j2=47
      go to 10
 50   j1=50
      j2=56
      go to 10
 60   j1=60
```

```
      j2=66
      go to 10
 70   continue
c
c     Print results
      call prmat(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *           sm,lacc,fsml,fsm2,fsm3,fsm4,fsm5,fsm6,asml,asm2,asm3,
     *           asm4,asm5,asm6,
     *           psml,psm2,psm3,psm4,psm5,psm6)
      return
      end
      subroutine prmat(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *                sm,lacc,fsm1,fsm2,fsm3,fsm4,fsm5,fsm6,asm1,
     *                asm2,asm3,asm4,asm5,asm6,
     *                psm1,psm2,psm3,psm4,psm5,psm6)
c*********************************************************************
c*********************************************************************
c*                                                                  **
c*    PRMAT                                                         **
c*                                                                  **
c*    This subroutine prints the material factors just calculated by **
c*      subroutine CALMAT.                                          **
c*                                                                  **
c*                                                                  **
c*********************************************************************
c*********************************************************************
      integer sm
      character uact*20,dact*40
      dimension sm(6),lacc(66),fsm1(66),fsm2(66),fsm3(66),fsm4(66),
     *          fsm5(66),fsm6(66),asm1(66),asm2(66),asm3(66),asm4(66),
     *          asm5(66),asm6(66),psm1(66),psm2(66),psm3(66),psm4(66),
     *          psm5(66),psm6(66)
c
c     Print title
      write(3,905)iact,dact,ibm,iby,iem,iey,uact
 905  format(1h1,39hROUTINE MAINTENANCE REPORT FOR ACTIVITY,1x,i3,2x,
     *       a40,5x,4hFROM,1x,i2,1h-,i2,1x,7hTHROUGH,1x,i2,1h-,i2,
     *       /,21x,20hACCOMPLISHMENT UNIT:,5x,a20)
c
c     Check to see if printing interstate, other state highway, or
c       total system information and print appropriate title
      if(iclass.ne.0)go to 100
      write(3,906)
 906  format(/,1x,31hINTERSTATE MATERIAL INFORMATION)
      go to 120
 100  if(iclass.ne.1)go to 110
      write(3,907)
 907  format(/,1x,40hOTHER STATE HIGHWAY MATERIAL INFORMATION)
      go to 120
 110  continue
      write(3,908)
 908  format(/,1x,38hTOTAL (INT + OSH) MATERIAL INFORMATION)
 120  continue
c     Print column headings
      write(3,900) (sm(j),j=1,6)
 900  format(///,6x,4hCREW,1x,6(1x,8h-----MAT,1x,i4,5h-----))
      write(3,902)
 902  format(1x,4hUNIT,1x,4hDAYS,1x,
     *  6(1x,4hFRAC,1x,6hAVGQNT,1x,6hQNT/AC))
      write(3,903)
 903  format(1x,4h----,1x,5h-----,
     *  6(1x,4h----,1x,6h------,1x,6h------)//)
c
c     Write the data
```

```
      j1=10
      j2=16
 10   do 11 j=j1,j2
      write(3,904)j,lacc(j),fsm1(j),asm1(j),psm1(j),fsm2(j),asm2(j),
     *            psm2(j),fsm3(j),asm3(j),psm3(j),fsm4(j),asm4(j),
     *            psm4(j),fsm5(j),asm5(j),psm5(j),fsm6(j),asm6(j),
     *            psm6(j)
 904  format(1x,i2,2h00,1x,i5,6(1x,f4.2,1x,f6.2,1x,f6.2))
 11   continue
      if(j1.eq.10)go to 20
      if(j1.eq.20)go to 30
      if(j1.eq.30)go to 40
      if(j1.eq.40)go to 50
      if(j1.eq.50)go to 60
      if(j1.eq.60)go to 70
 20   j1=20
      j2=26
      go to 10
 30   j1=30
      j2=36
      go to 10
 40   j1=40
      j2=47
      go to 10
 50   j1=50
      j2=56
      go to 10
 60   j1=60
      j2=66
      go to 10
 70   continue
      return
      end
      subroutine cost(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *                sm,csm,crh,cot,lacc,tacc,arh,aot,frh,fot,
     *                fsm1,fsm2,fsm3,fsm4,fsm5,fsm6,asm1,asm2,
     *                asm3,asm4,asm5,asm6,actot)
c***********************************************************************
c***********************************************************************
c*                                                                   **
c*    COST                                                           **
c*                                                                   **
c*    Using information from other routines, the average cost per    **
c*      accomplishment unit for each subdistrict is calculated here. **
c*      The cost is for labor and materials only.                    **
c*                                                                   **
c*    The following information is provided by other subroutines:    **
c*                                                                   **
c*      lacc    -- the total number of times the activity was perform- **
c*                 ed                                                **
c*      tacc    -- total accomplishment (amount of work done)        **
c*      arh     -- average regular hours per crew day = trh / lrh    **
c*      frh     -- fraction of time regular-time hours were used     **
```

```
c*                = lrh / lacc                                         **
c*     crh    — regular labor wage per hour                            **
c*     aot    — average overtime hours per crew day = tot / lot        **
c*     fot    — fraction of time overtime hours were used              **
c*                = lot / lacc                                         **
c*     cot    — overtime labor wage per hour                           **
c*     fsm1   — fraction of time specified material 1 is used          **
c*                = lsm1 / lacc                                        **
c*     asm1   — average quantity of specified material 1 used per      **
c*                crew day when that material is used = qsm1 / lsm1    **
c*     csm(1) — unit cost for specified material 1                     **
c*     These four factors, fsm_, asm_, psm_, and csm are repeated      **
c*               for each specified material, 1 through 6.             **
c*                                                                     **
c*    Cost is calculated as follows:                                   **
c*                                                                     **
c*    Resource cost per unit of accomplishment =                       **
c*      (# of records * Fraction of time resource is used *            **
c*       Avg quantity of resource when used * Unit cost of resource)   **
c*       / Total accomplishment                                        **
c*     The following cost figures are calculated:                      **
c*                                                                     **
c*     acrh    — average cost for regular labor = (lacc * frh  *       **
c*                arh * crh) / tacc                                    **
c*     acot    — average cost for overtime labor = (lacc * fot *       **
c*                aot * cot) / tacc                                    **
c*     acsm1   — average cost for specified material 1 = (lacc *       **
c*                fsm1 * asm1 * csm(1)) / tacc                         **
c*             This material cost calculation is made for each speci- **
c*             fied material, 1 through 6.                             **
c*     NOTE:  In the above calculations, 0.0000000001 is added to      **
c*            denominator to avoid a possible floating divide by zero  **
c*            error.                                                   **
c*                                                                     **
c*     aclab   — average cost for labor = acrh + acot                  **
c*     acmat   — average cost for materials = acsm1 + acsm2 + ... +    **
c*                acsm6                                                **
c*     actot   — average total cost for materials and labor =          **
c*                aclab + acmat                                        **
c*                                                                     **
c*                                                                     **
c***********************************************************************
c***********************************************************************
      integer sm
      character uact*20,dact*40
      dimension sm(6),csm(6),lacc(66),tacc(66),arh(66),aot(66),
     *          frh(66),fot(66),fsm1(66),fsm2(66),fsm3(66),fsm4(66),
     *          fsm5(66),fsm6(66),asm1(66),asm2(66),asm3(66),asm4(66),
     *          asm5(66),asm6(66),
     *          acrh(66),acot(66),acsm1(66),acsm2(66),acsm3(66),
     *          acsm4(66),acsm5(66),acsm6(66),aclab(66),acmat(66),
     *          actot(66)
c
```

```
c     Calculate cost factors for each subdistrict
      j1=10
      j2=16
 10   do 11 j=j1,j2
      acrh(j)=(float(lacc(j)) * frh(j) * arh(j) * crh) / (tacc(j)
     *          + .0000000001)
      acot(j)=(float(lacc(j)) * fot(j) * aot(j) * cot) / (tacc(j)
     *          + .0000000001)
      acsm1(j)=(float(lacc(j)) * fsm1(j) * asm1(j) * csm(1)) / (tacc(j)
     *          + .0000000001)
      acsm2(j)=(float(lacc(j)) * fsm2(j) * asm2(j) * csm(2)) / (tacc(j)
     *          + .0000000001)
      acsm3(j)=(float(lacc(j)) * fsm3(j) * asm3(j) * csm(3)) / (tacc(j)
     *          + .0000000001)
      acsm4(j)=(float(lacc(j)) * fsm4(j) * asm4(j) * csm(4)) / (tacc(j)
     *          + .0000000001)
      acsm5(j)=(float(lacc(j)) * fsm5(j) * asm5(j) * csm(5)) / (tacc(j)
     *          + .0000000001)
      acsm6(j)=(float(lacc(j)) * fsm6(j) * asm6(j) * csm(6)) / (tacc(j)
     *          + .0000000001)
      aclab(j)=acrh(j) + acot(j)
      acmat(j)=acsm1(j) + acsm2(j) + acsm3(j) + acsm4(j) + acsm5(j)
     *          + acsm6(j)
      actot(j)=aclab(j) + acmat(j)
 11   continue
      if(j1.eq.10)go to 20
      if(j1.eq.20)go to 30
      if(j1.eq.30)go to 40
      if(j1.eq.40)go to 50
      if(j1.eq.50)go to 60
      if(j1.eq.60)go to 70
 20   j1=20
      j2=26
      go to 10
 30   j1=30
      j2=36
      go to 10
 40   j1=40
      j2=47
      go to 10
 50   j1=50
      j2=56
      go to 10
 60   j1=60
      j2=66
      go to 10
 70   continue
c
c      Print cost data
      call prcost(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *            sm,lacc,tacc,acrh,acot,acsm1,acsm2,acsm3,acsm4,
     *            acsm5,acsm6,aclab,acmat,actot)
      return
      end
      subroutine prcost(iact,dact,uact,ibm,iby,iem,iey,iclass,
     *                  sm,lacc,tacc,acrh,acot,acsm1,acsm2,acsm3,
     *                  acsm4,acsm5,acsm6,aclab,acmat,actot)
c***********************************************************************
c***********************************************************************
c*                                                                   **
c*    PRCOST                                                         **
c*                                                                   **
c*    This subroutine prints the cost factors just calculated by sub- **
c*      routine COST.                                                **
c*                                                                   **
c*                                                                   **
c***********************************************************************
c***********************************************************************
      integer sm
      character uact*20,dact*40
      dimension sm(6),lacc(66),tacc(66),acrh(66),acot(66),acsm1(66),
     *          acsm2(66),acsm3(66),acsm4(66),acsm5(66),acsm6(66),
     *          actot(66),aclab(66),acmat(66)
c
c     Print title
      write(3,904)iact,dact,ibm,iby,iem,iey,uact
 904  format(1h1,39hROUTINE MAINTENANCE REPORT FOR ACTIVITY,1x,i3,2x,
     *       a40,5x,4hFROM,1x,i2,1h-,i2,1x,7hTHROUGH,1x,i2,1h-,i2,
     *       /,21x,20hACCOMPLISHMENT UNIT:,5x,a20)
c
c     Check to see which highway system the data is for, and print
c       appropriate heading
      if(iclass.ne.0)go to 100
      write(3,905)
 905  format(/,1x,27hINTERSTATE COST INFORMATION,5x,
     *       50h(ALL COSTS ARE DOLLARS PER UNIT OF ACCOMPLISHMENT))
      go to 120
 100  continue
      if(iclass.ne.1)go to 110
      write(3,906)
 906  format(/,1x,36hOTHER STATE HIGHWAY COST INFORMATION,5x,
     *       50h(ALL COSTS ARE DOLLARS PER UNIT OF ACCOMPLISHMENT))
      go to 120
 110  continue
      write(3,907)
 907  format(/,1x,34hTOTAL (INT + OSH) COST INFORMATION,5x,
     *       50h(ALL COSTS ARE DOLLARS PER UNIT OF ACCOMPLISHMENT))
 120  continue
c     Print column headings
      write(3,900)
 900  format(//,8x,4hCREW,5x,5hTOTAL,33x,18(1h-),17hCOSTS BY MATERIAL,
     *       23(1h-))
      write(3,901)(sm(j),j=1,6)
 901  format(1x,4hUNIT,3x,4hDAYS,5x,6hACCOMP,2x,8hTOT COST,2x,
     *       8hLAB COST,2x,8hMAT COST,2x,6(2x,i4,4x))
      write(3,902)
 902   format(1x,4h----,2x,6h------,4x,6h------,9(2x,8h--------),//)
c
c      Print cost figures for each subdistrict
       j1=10
       j2=16
 10    do 11 j=j1,j2
       write(3,903)j,lacc(j),tacc(j),actot(j),aclab(j),acmat(j),
      *             acsm1(j),acsm2(j),acsm3(j),acsm4(j),acsm5(j),acsm6(j)
 903   format(1x,i2,2h00,2x,i6,2x,f8.1,9(2x,f8.3))
 11    continue
       if(j1.eq.10)go to 20
       if(j1.eq.20)go to 30
       if(j1.eq.30)go to 40
       if(j1.eq.40)go to 50
       if(j1.eq.50)go to 60
       if(j1.eq.60)go to 70
 20    j1=20
       j2=26
       go to 10
 30    j1=30
       j2=36
       go to 10
 40    j1=40
       j2=47
       go to 10
 50    j1=50
       j2=56
       go to 10
 60    j1=60
       j2=66
       go to 10
 70    continue
       return
       end
       subroutine dstsum(iact,dact,uact,ibm,iby,iem,iey,iclass,
      *           tacc,tcrew,trh,tot,rhacc,otacc,lacc,lrh,lot,arh,
      *           frh,aot,fot,sm,qsm1,qsm2,qsm3,qsm4,qsm5,qsm6,
      *           lsm1,lsm2,lsm3,lsm4,lsm5,lsm6,sm1ac,sm2ac,sm3ac,
      *           sm4ac,sm5ac,sm6ac,csm,crh,cot)
c*********************************************************************
c*********************************************************************
c*                                                                  **
c*     DSTSUM                                                       **
c*                                                                  **
c*     This subroutine calls other subroutines to calculate and print **
c*       summaries of the labor, material, and cost information for **
c*       each district and for the state as a whole.                **
c*                                                                  **
c*                                                                  **
c*********************************************************************
c*********************************************************************

       integer tcrew,trh,tot,sm
       character uact*20,dact*40
      dimension tacc(66),tcrew(66),trh(66),tot(66),rhacc(66),
     *          otacc(66),lacc(66),lrh(66),lot(66),arh(66),frh(66),
     *          aot(66),fot(66),sm(6),qsm1(66),qsm2(66),qsm3(66),
     *          qsm4(66),qsm5(66),qsm6(66),lsm1(66),lsm2(66),lsm3(66),
     *          lsm4(66),lsm5(66),lsm6(66),sm1ac(66),sm2ac(66),
     *          sm3ac(66),sm4ac(66),sm5ac(66),sm6ac(66),csm(6),
     *          ldacc(6),tdacc(6),adacc(6),adcrw(6),ldrh(6),dprh(6),
     *          adrh(6),dfrh(6),ldot(6),dpot(6),adot(6),dfot(6),
     *          dfsm1(6),dfsm2(6),dfsm3(6),dfsm4(6),dfsm5(6),dfsm6(6),
     *          dasm1(6),dasm2(6),dasm3(6),dasm4(6),dasm5(6),dasm6(6),
     *          dpsm1(6),dpsm2(6),dpsm3(6),dpsm4(6),dpsm5(6),dpsm6(6),
     *          dacrh(6),dacot(6),dacsm1(6),dacsm2(6),dacsm3(6),
     *          dacsm4(6),dacsm5(6),dacsm6(6),daclab(6),dacmat(6),
     *          dactot(6)
c
c     First calculate district and state labor summary
      call dstlab(tacc,tcrew,trh,tot,rhacc,otacc,lacc,lrh,lot,
     *     ldacc,tdacc,adacc,adcrw,ldrh,dprh,adrh,dfrh,ldot,
     *     dpot,adot,dfot,lsacc,tsacc,asacc,ascrw,lsrh,sprh,asrh,sfrh,
     *     lsot,spot,asot,sfot)
c
c     Calculate district and state material summary
      call dstmat(lacc,ldacc,lsacc,qsm1,qsm2,qsm3,qsm4,qsm5,qsm6,
     *            lsm1,lsm2,lsm3,lsm4,lsm5,lsm6,sm1ac,sm2ac,sm3ac,
     *            sm4ac,sm5ac,sm6ac,dfsm1,dfsm2,dfsm3,dfsm4,dfsm5,
     *            dfsm6,dasm1,dasm2,dasm3,dasm4,dasm5,dasm6,sfsm1,
     *            sfsm2,sfsm3,sfsm4,sfsm5,sfsm6,sasm1,sasm2,sasm3,
     *            sasm4,sasm5,sasm6,dpsm1,dpsm2,dpsm3,dpsm4,dpsm5,
     *            dpsm6,spsm1,spsm2,spsm3,spsm4,spsm5,spsm6)
c
c     Calculate district and state cost summary
      call dstcst(csm,crh,cot,ldacc,tdacc,adrh,adot,dfrh,dfot,
     *            dfsm1,dfsm2,dfsm3,dfsm4,dfsm5,dfsm6,dasm1,dasm2,
     *            dasm3,dasm4,dasm5,dasm6,lsacc,tsacc,asrh,asot,sfrh,
     *            sfot,sfsm1,sfsm2,sfsm3,sfsm4,sfsm5,sfsm6,sasm1,sasm2,
     *            sasm3,sasm4,sasm5,sasm6,dacrh,dacot,dacsm1,dacsm2,
     *            dacsm3,dacsm4,dacsm5,dacsm6,daclab,dacmat,dactot,
     *            sacrh,sacot,sacsm1,sacsm2,sacsm3,sacsm4,sacsm5,
     *            sacsm6,saclab,sacmat,sactot)
c
c     Print district summary
      call pdist(iact,dact,uact,ibm,iby,iem,iey,iclass,ldacc,
     *           tdacc,adacc,adcrw,ldrh,dprh,adrh,dfrh,ldot,
     *           dpot,adot,dfot,sm,dfsm1,dfsm2,dfsm3,dfsm4,dfsm5,
     *           dfsm6,dasm1,dasm2,dasm3,dasm4,dasm5,dasm6,dpsm1,
     *           dpsm2,dpsm3,dpsm4,dpsm5,dpsm6,dacrh,dacot,dacsm1,
     *           dacsm2,dacsm3,dacsm4,dacsm5,dacsm6,daclab,dacmat,
     *           dactot)
c
c     Print state summary
      call pstate(iact,dact,uact,ibm,iby,iem,iey,iclass,lsacc,
     *           tsacc,asacc,ascrw,lsrh,sprh,asrh,sfrh,lsot,
     *           spot,asot,sfot,sm,sfsm1,sfsm2,sfsm3,sfsm4,sfsm5,
     *           sfsm6,sasml,sasm2,sasm3,sasm4,sasm5,sasm6,spsml,
     *           spsm2,spsm3,spsm4,spsm5,spsm6,sacrh,sacot,sacsml,
     *           sacsm2,sacsm3,sacsm4,sacsm5,sacsm6,saclab,sacmat,
     *           sactot)
      return
      end
      subroutine dstlab(tacc,tcrew,trh,tot,rhacc,otacc,lacc,lrh,
     *                 lot,ldacc,dtacc,adacc,adcrw,ldrh,dprh,adrh,
     *                 dfrh,ldot,dpot,adot,dfot,lsacc,tsacc,asacc,
     *                 ascrw,lsrh,sprh,asrh,sfrh,lsot,spot,asot,sfot)
c**********************************************************************
c**********************************************************************
c*                                                                   **
c*    DSTLAB                                                         **
c*                                                                   **
c*    This subroutine summarizes labor information for each district **
c*      and for the state as a whole.                                **
c*                                                                   **
c*    The following values for each subdistrict are provided from    **
c*      other subroutines:                                           **
c*                                                                   **
c*      lacc  -- the total number of times the activity was performed **
c*      tacc  -- total accomplishment (amount of work done)          **
c*      tcrew -- total of crew sizes in persons                      **
c*      lrh   -- number of times regular-time labor hours were       **
c*               used in performance of the activity                 **
c*      trh   -- total number of regular-time labor hours used       **
c*      rhacc -- total accomplishment when regular-time labor hours  **
c*               were used                                           **
c*      lot   -- number of time overtime labor hours were used       **
c*      tot   -- total number of overtime labor hours used           **
c*      otacc -- total accomplishment when overtime labor hours were **
c*               used                                                **
c*                                                                   **
c*    The following values are calculated for each of the six IDOH   **
c*      districts:                                                   **
c*                                                                   **
c*      ldacc -- total number of times activity was performed by the **
c*               district = sum of lacc for each district            **
c*      dtacc -- district total accomplishment = sum of tacc for     **
c*               each district                                       **
c*      dcrew -- total of crews in a district = sum of tcrew for     **
c*               each district                                       **
c*      ldrh  -- total number of times regular-time labor hours were **
c*               reported = sum of lrh for a district                **
c*      dtrh  -- district total regular-time hours used = sum of trh **
c*               for a district                                      **
c*      drhac -- district accomplishment when regular hours were     **
c*               used = sum of rhacc for a district                  **
c*      ldot  -- same as ldrh but with overtime hours                **
c*      dtot  -- same as dtrh but with overtime hours                **
c*      dotac -- same as drhac but with overtime hours               **
c*                                                                   **
```

```
c*   A similar set of values is tabulated for the state as a whole:  **
c*                                                                   **
c*     lsacc — total number of times activity was performed in the  **
c*             state = sum of all lacc values                        **
c*     tsacc — total state accomplishment = sum of all tacc values   **
c*     screw — total of state crews = sum of all tcrew values        **
c*     lsrh  — total number of times regular-times hours were used   **
c*             in the state = sum of all lrh values                  **
c*     strh  — state total number of regular-time hours used         **
c*             = sum of all trh values                               **
c*     srhac — total state accomplishment when regular hours were    **
c*             reported = sum of all rhacc values                    **
c*     lsot  — same as lsrh but with overtime hours                  **
c*     stot  — same as strh but with overtime hours                  **
c*     sotac — same as srhac but with overtime hours                 **
c*                                                                   **
c*   The following factors are calculated for each IDOH district:    **
c*                                                                   **
c*     adacc — average district accomplishment = dtacc / ldacc       **
c*     adcrw — average district crew size = dcrew / ldacc            **
c*     adrh  — average district regular hours = dtrh / ldrh          **
c*     adrac — average district regular hour accomplishment          **
c*             = drhac / ldrh                                        **
c*     dfrh  — district fraction of the time regular hours was used **
c*             = ldrh / ldacc                                        **
c*     dprh  — district regular hour productivity = dtrh / drhac     **
c*     adot  — average district overtime = doth / ldot               **
c*     adoac — average district overtime accomplishment              **
c*             = dotac / ldot                                        **
c*     dfot  — district fraction of the time overtime was used       **
c*             = ldot / ldacc                                        **
c*     dpot  — district overtime productivity = dtot / dotac         **
c*                                                                   **
c*   Similar factors are calculated for the state as a whole:        **
c*                                                                   **
c*     asacc — average state accomplishment = tsacc / lsacc          **
c*     ascrw — average state crew size = screw / lsacc               **
c*     asrh  — average state regular hours = strh / lsrh             **
c*     asrac — average state regular hour accomplishment             **
c*             = srhac / lsrh                                        **
c*     sfrh  — state fraction of the time regular hours was used     **
c*             = lsrh / lsacc                                        **
c*     sprh  — state regular hour productivity = strh / srhac        **
c*     asot  — average state overtime = soth / lsot                  **
c*     asoac — average state overtime accomplishment                 **
c*             = sotac / lsot                                        **
c*     sfot  -- state fraction of the time overtime was used         **
c*             = lsot / lsacc                                        **
c*     spot  — state overtime productivity = stot / sotac            **
c*                                                                   **
c*   NOTE: In the calculations listed in these last two sections,    **
c*         0.0000000001 is added to each denominator to avoid a      **
c*         possible floating divide by zero error.                   **
```

```
c*                                                                    **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      integer tcrew,trh,tot,dcrew,dtrh,dtot,screw,strh,stot
      dimension tacc(66),tcrew(66),trh(66),tot(66),rhacc(66),
     *          otacc(66),lacc(66),lrh(66),lot(66),dtacc(6),dcrew(6),
     *          dtrh(6),dtot(6),drhac(6),dotac(6),ldacc(6),ldrh(6),
     *          ldot(6),adrh(6),dfrh(6),adot(6),dfot(6),dprh(6),
     *          dpot(6),adacc(6),adcrw(6),adrac(6),adoac(6)
c
c     Because this subroutine is called separately to calculate the
c       interstate, other state highway, and total system figures,
c       the district and state summations must be set to zero each
c       time the subroutine is used.
      tsacc=0.
      screw=0
      strh=0
      stot=0
      srhac=0.
      sotac=0.
      lsrh=0
      lsot=0
      lsacc=0
      do 1 k=1,6
      dtacc(k)=0.
      dcrew(k)=0
      dtrh(k)=0
      dtot(k)=0
      drhac(k)=0.
      dotac(k)=0.
      ldrh(k)=0
      ldot(k)=0
      ldacc(k)=0
 1    continue
c
c     Make the desired summations for each district, and the entire
c       state.  Counter j tracks the subdistricts, while counter k
c       tracks the six districts.
      j1=10
      j2=16
      k=1
 10   do 11 j=j1,j2
      dtacc(k)=dtacc(k)+tacc(j)
      tsacc=tsacc+tacc(j)
      dcrew(k)=dcrew(k)+tcrew(j)
      screw=screw+tcrew(j)
      dtrh(k)=dtrh(k)+trh(j)
      strh=strh+trh(j)
      dtot(k)=dtot(k)+tot(j)
      stot=stot+tot(j)
      drhac(k)=drhac(k)+rhacc(j)
      srhac=srhac+rhacc(j)
      dotac(k)=dotac(k)+otacc(j)
      sotac=sotac+otacc(j)
      ldacc(k)=ldacc(k)+lacc(j)
      lsacc=lsacc+lacc(j)
      ldrh(k)=ldrh(k)+lrh(j)
      lsrh=lsrh+lrh(j)
      ldot(k)=ldot(k)+lot(j)
      lsot=lsot+lot(j)
 11   continue
      if(k.eq.1)go to 20
      if(k.eq.2)go to 30
      if(k.eq.3)go to 40
      if(k.eq.4)go to 50
      if(k.eq.5)go to 60
      if(k.eq.6)go to 70
 20   j1=20
      j2=26
      k=2
      go to 10
 30   j1=30
      j2=36
      k=3
      go to 10
 40   j1=40
      j2=47
      k=4
      go to 10
 50   j1=50
      j2=56
      k=5
      go to 10
 60   j1=60
      j2=66
      k=6
      go to 10
 70   continue
c
c     Calculate the various factors for each of the six districts
 100  do 111 k=1,6
      adacc(k)=dtacc(k) / (float(ldacc(k)) + .0000000001)
      adcrw(k)=float(dcrew(k)) / (float(ldacc(k)) + .0000000001)
      adrh(k)=float(dtrh(k)) / (float(ldrh(k)) + .0000000001)
      adrac(k)=drhac(k) / (float(ldrh(k)) + .0000000001)
      dfrh(k)=float(ldrh(k)) / (float(ldacc(k)) + .0000000001)
      adot(k)=float(dtot(k)) / (float(ldot(k)) + .0000000001)
      adoac(k)=dotac(k) / (float(ldot(k)) + .0000000001)
      dfot(k)=float(ldot(k)) / (float(ldacc(k)) + .0000000001)
      dprh(k)=float(dtrh(k)) / (drhac(k) + .0000000001)
      dpot(k)=float(dtot(k)) / (dotac(k) + .0000000001)
 111  continue
c
c     Calculate state factors
      asacc=tsacc / (float(lsacc) + .0000000001)
      ascrw=float(screw) / (float(lsacc) + .0000000001)
      asrh=float(strh) / (float(lsrh) + .0000000001)
      asrac=srhac / (float(lsrh) + .0000000001)
      sfrh=float(lsrh) / (float(lsacc) + .0000000001)
      asot=float(stot) / (float(lsot) + .0000000001)
      asoac=sotac / (float(lsot) + .0000000001)
      sfot=float(lsot) / (float(lsacc) + .0000000001)
      sprh=float(strh) / (srhac + .0000000001)
      spot=float(stot) / (sotac + .0000000001)
      return
      end
      subroutine dstmat(lacc,ldacc,lsacc,qsml,qsm2,qsm3,qsm4,qsm5,qsm6,
     *                  lsml,lsm2,lsm3,lsm4,lsm5,lsm6,smlac,sm2ac,sm3ac,
     *                  sm4ac,sm5ac,sm6ac,dfsml,dfsm2,dfsm3,dfsm4,
     *                  dfsm5,dfsm6,dasml,dasm2,dasm3,dasm4,dasm5,dasm6,
     *                  sfsml,sfsm2,sfsm3,sfsm4,sfsm5,sfsm6,sasml,sasm2,
     *                  sasm3,sasm4,sasm5,sasm6,dpsml,dpsm2,dpsm3,dpsm4,
     *                  dpsm5,dpsm6,spsml,spsm2,spsm3,spsm4,spsm5,spsm6)
c***********************************************************************
c***********************************************************************
c*                                                                   **
c*    DSTMAT                                                         **
c*                                                                   **
c*    This subroutine summarizes material information for each dis-  **
c*      trict, and for the state as a whole.                         **
c*                                                                   **
c*    The following values for each subdisrict are provided from     **
c*      other subroutines:                                           **
c*                                                                   **
c*     qsml  -- total amount of specified material 1 used (specified **
c*              material 1 is the first material listed in the input **
c*              parameter file)                                      **
c*     lsml  -- total number of times specified material 1 was used  **
c*     smlac -- total work accomplished when specified material 1 was **
c*              used                                                 **
c*     These three variables, qsm_, lsm_, and am_ac, are repeated for **
c*              for each specified material, 1 through 6.            **
c*     ldacc -- total number of times the activity was performed by  **
c*              each district                                        **
c*
c*   Using this information, the following factors are calculated:   **
c*                                                                   **
c*     dqsml -- total district amount of specified material 1 used   **
c*              (specified material 1 is the first material listed in **
c*              the input parameter file) = sum of qsml for each     **
c*              district                                             **
c*     ldsml -- total number of times specified material 1 was used  **
c*              in the district = sum of lsml for each district      **
c*     dmlac -- total district work accomplished when specified      **
c*              material 1 was used = sum of smlac for each district **
c*                                                                   **
c*     sqsml -- total state amount of specified material 1 used      **
c*              (specified material 1 is the first material listed in **
c*              the input parameter file) = sum of all qsm1 values      **
c*     lssm1 — total number of times specified material 1 was used      **
c*             in the state = sum of all lsm1 values                    **
c*     sacm1 — total state work accomplished when specified material   **
c*             1 was used = sum of all sm1ac values                     **
c*                                                                      **
c*     These six variables are repeated for each specified material,    **
c*             1 through 6.                                             **
c*                                                                      **
c*   The following factors are calculated for each IDOH district:       **
c*                                                                      **
c*     dfsm1 — fraction of time specified material 1 is used            **
c*             = ldsm1 / ldacc                                          **
c*     dasm1 — average quantity of specified material 1 used per        **
c*             crew day when that material is used = dqsm1 / ldsm1      **
c*     dpsm1 — district productivity with specified material 1          **
c*             = dqsm1 / dm1ac                                          **
c*                                                                      **
c*     These three factors, dfsm_, dasm_, and dpsm_, are repeated for  **
c*             each specified material, 1 through 6.                    **
c*                                                                      **
c*     Similar calculations are made for the state as a whole:          **
c*                                                                      **
c*     sfsm1 — fraction of time specified material 1 is used            **
c*             = lssm1 / lsacc                                          **
c*     sasm1 — average quantity of specified material 1 used per        **
c*             crew day when that material is used = sqsm1 / lssm1      **
c*     spsm1 — state productivity with specified material 1             **
c*             = sqsm1 / sacm1                                          **
c*                                                                      **
c*     These three factors, sfsm_, sasm_, and spsm_, are repeated for  **
c*             each specified material, 1 through 6.                    **
c*                                                                      **
c*     NOTE:  In calculating the factors in the last two sections,      **
c*            0.0000000001 is added to each denominator to avoid a      **
c*            floating divide by zero error.                            **
c*                                                                      **
c*                                                                      **
c***********************************************************************
c***********************************************************************
      dimension lacc(66),qsm1(66),qsm2(66),qsm3(66),qsm4(66),
     *          qsm5(66),qsm6(66),lsm1(66),lsm2(66),lsm3(66),lsm4(66),
     *          lsm5(66),lsm6(66),ldacc(6),dqsm1(6),dqsm2(6),dqsm3(6),
     *          dqsm4(6),dqsm5(6),dqsm6(6),ldsm1(6),ldsm2(6),ldsm3(6),
     *          ldsm4(6),ldsm5(6),ldsm6(6),dfsm1(6),dfsm2(6),dfsm3(6),
     *          dfsm4(6),dfsm5(6),dfsm6(6),dasm1(6),dasm2(6),dasm3(6),
     *          dasm4(6),dasm5(6),dasm6(6),
     *          sm1ac(66),sm2ac(66),sm3ac(66),sm4ac(66),sm5ac(66),
     *          sm6ac(66),dm1ac(6),dm2ac(6),dm3ac(6),dm4ac(6),
     *          dm5ac(6),dm6ac(6),dpsm1(6),dpsm2(6),dpsm3(6),
     *          dpsm4(6),dpsm5(6),dpsm6(6)
c
c     Because this subroutine is called separately to calculate the
```

```
c         interstate, other state highway, and total system figures,
c         the district and state summations must be set to zero each
c         time the subroutine is used.
c
c     Set state summations equal to zero
      sqsm1=0.
      lssm1=0
      sacm1=0.
      sqsm2=0.
      lssm2=0
      sacm2=0.
      sqsm3=0.
      lssm3=0
      sacm3=0.
      sqsm4=0.
      lssm4=0
      sacm4=0.
      sqsm5=0.
      lssm5=0
      sacm5=0.
      sqsm6=0.
      lssm6=0
      sacm6=0.
c
c     Set summations for the six districts equal to zero
      do 1 k=1,6
      dqsm1(k)=0.
      ldsm1(k)=0
      dm1ac(k)=0.
      dqsm2(k)=0.
      ldsm2(k)=0
      dm2ac(k)=0.
      dqsm3(k)=0.
      ldsm3(k)=0
      dm3ac(k)=0.
      dqsm4(k)=0.
      ldsm4(k)=0
      dm4ac(k)=0.
      dqsm5(k)=0.
      ldsm5(k)=0
      dm5ac(k)=0.
      dqsm6(k)=0.
      ldsm6(k)=0
      dm6ac(k)=0.
 1    continue
c
c     Make the desired summations for ech district, and the entire
c       state.  Counter j tracksthe subdistricts, while counter k
c       tracks the six districts.
      j1=10
      j2=16
      k=1
 10   do 11 j=j1,j2
```

```
      dqsm1(k)=dqsm1(k)+qsm1(j)
      ldsm1(k)=ldsm1(k)+lsm1(j)
      sqsm1=sqsm1+qsm1(j)
      lssm1=lssm1+lsm1(j)
      dm1ac(k)=dm1ac(k)+sm1ac(j)
      sacm1=sacm1+sm1ac(j)
      dqsm2(k)=dqsm2(k)+qsm2(j)
      ldsm2(k)=ldsm2(k)+lsm2(j)
      sqsm2=sqsm2+qsm2(j)
      lssm2=lssm2+lsm2(j)
      dm2ac(k)=dm2ac(k)+sm2ac(j)
      sacm2=sacm2+sm2ac(j)
      dqsm3(k)=dqsm3(k)+qsm3(j)
      ldsm3(k)=ldsm3(k)+lsm3(j)
      sqsm3=sqsm3+qsm3(j)
      lssm3=lssm3+lsm3(j)
      dm3ac(k)=dm3ac(k)+sm3ac(j)
      sacm3=sacm3+sm3ac(j)
      dqsm4(k)=dqsm4(k)+qsm4(j)
      ldsm4(k)=ldsm4(k)+lsm4(j)
      sqsm4=sqsm4+qsm4(j)
      lssm4=lssm4+lsm4(j)
      dm4ac(k)=dm4ac(k)+sm4ac(j)
      sacm4=sacm4+sm4ac(j)
      dqsm5(k)=dqsm5(k)+qsm5(j)
      ldsm5(k)=ldsm5(k)+lsm5(j)
      sqsm5=sqsm5+qsm5(j)
      lssm5=lssm5+lsm5(j)
      dm5ac(k)=dm5ac(k)+sm5ac(j)
      sacm5=sacm5+sm5ac(j)
      dqsm6(k)=dqsm6(k)+qsm6(j)
      ldsm6(k)=ldsm6(k)+lsm6(j)
      sqsm6=sqsm6+qsm6(j)
      lssm6=lssm6+lsm6(j)
      dm6ac(k)=dm6ac(k)+sm6ac(j)
      sacm6=sacm6+sm6ac(j)
 11   continue
      if(k.eq.1)go to 20
      if(k.eq.2)go to 30
      if(k.eq.3)go to 40
      if(k.eq.4)go to 50
      if(k.eq.5)go to 60
      if(k.eq.6)go to 70
 20   j1=20
      j2=26
      k=2
      go to 10
 30   j1=30
      j2=36
      k=3
      go to 10
 40   j1=40
      j2=47
      k=4
      go to 10
 50   j1=50
      j2=56
      k=5
      go to 10
 60   j1=60
      j2=66
      k=6
      go to 10
 70   continue
c
c     Calculate the various factors for each of the six districts.
 100  do 111 k=1,6
      dfsm1(k)=float(ldsm1(k)) / (float(ldacc(k)) + .0000000001)
      dfsm2(k)=float(ldsm2(k)) / (float(ldacc(k)) + .0000000001)
      dfsm3(k)=float(ldsm3(k)) / (float(ldacc(k)) + .0000000001)
      dfsm4(k)=float(ldsm4(k)) / (float(ldacc(k)) + .0000000001)
      dfsm5(k)=float(ldsm5(k)) / (float(ldacc(k)) + .0000000001)
      dfsm6(k)=float(ldsm6(k)) / (float(ldacc(k)) + .0000000001)
      dasm1(k)=dqsm1(k) / (float(ldsm1(k)) + .0000000001)
      dasm2(k)=dqsm2(k) / (float(ldsm2(k)) + .0000000001)
      dasm3(k)=dqsm3(k) / (float(ldsm3(k)) + .0000000001)
      dasm4(k)=dqsm4(k) / (float(ldsm4(k)) + .0000000001)
      dasm5(k)=dqsm5(k) / (float(ldsm5(k)) + .0000000001)
      dasm6(k)=dqsm6(k) / (float(ldsm6(k)) + .0000000001)
      dpsm1(k)=dqsm1(k) / (dm1ac(k) + .0000000001)
      dpsm2(k)=dqsm2(k) / (dm2ac(k) + .0000000001)
      dpsm3(k)=dqsm3(k) / (dm3ac(k) + .0000000001)
      dpsm4(k)=dqsm4(k) / (dm4ac(k) + .0000000001)
      dpsm5(k)=dqsm5(k) / (dm5ac(k) + .0000000001)
      dpsm6(k)=dqsm6(k) / (dm6ac(k) + .0000000001)
 111  continue
c
c     Calculate state factors
      sfsm1=float(lssm1) / (float(lsacc) + .0000000001)
      sfsm2=float(lssm2) / (float(lsacc) + .0000000001)
      sfsm3=float(lssm3) / (float(lsacc) + .0000000001)
      sfsm4=float(lssm4) / (float(lsacc) + .0000000001)
      sfsm5=float(lssm5) / (float(lsacc) + .0000000001)
      sfsm6=float(lssm6) / (float(lsacc) + .0000000001)
      sasm1=sqsm1 / (float(lssm1) + .0000000001)
      sasm2=sqsm2 / (float(lssm2) + .0000000001)
      sasm3=sqsm3 / (float(lssm3) + .0000000001)
      sasm4=sqsm4 / (float(lssm4) + .0000000001)
      sasm5=sqsm5 / (float(lssm5) + .0000000001)
      sasm6=sqsm6 / (float(lssm6) + .0000000001)
      spsm1=sqsm1 / (sacm1 + .0000000001)
      spsm2=sqsm2 / (sacm2 + .0000000001)
      spsm3=sqsm3 / (sacm3 + .0000000001)
      spsm4=sqsm4 / (sacm4 + .0000000001)
      spsm5=sqsm5 / (sacm5 + .0000000001)
      spsm6=sqsm6 / (sacm6 + .0000000001)
      return
      end
      subroutine dstcst(csm,crh,cot,ldacc,tdacc,adrh,adot,dfrh,
     *                  dfot,dfsm1,dfsm2,dfsm3,dfsm4,dfsm5,dfsm6,
     *                  dasm1,dasm2,dasm3,dasm4,dasm5,dasm6,lsacc,
     *                  tsacc,asrh,asot,sfrh,sfot,sfsm1,sfsm2,sfsm3,
     *                  sfsm4,sfsm5,sfsm6,sasm1,sasm2,sasm3,sasm4,
     *                  sasm5,sasm6,dacrh,dacot,dacsm1,dacsm2,dacsm3,
     *                  dscsm4,dacsm5,dacsm6,daclab,dacmat,dactot,
     *                  sacrh,sacot,sacsm1,sacsm2,sacsm3,sacsm4,
     *                  sacsm5,sacsm6,saclab,sacmat,sactot)
c**********************************************************************
c**********************************************************************
c*                                                                  **
c*    DSTCST                                                        **
c*                                                                  **
c*    Using information from other routines, the average cost per   **
c*      accomplishment unit for each district and the state as a    **
c*      whole are calculated here.  The cost is for labor and       **
c*      materials only.                                             **
c*                                                                  **
c*    The following information is provided by other subroutines:   **
c*                                                                  **
c*      ldacc  — the total number of times the activity was perform- **
c*               ed in each district                                 **
c*      tdacc  — total district accomplishment (amount of work done) **
c*      adrh   — average district regular hours per crew day        **
c*      dfrh   -- fraction of time regular-time hours were used     **
c*      crh    — regular labor wage per hour                        **
c*      adot   — average district overtime hours per crew day       **
c*      dfot   — fraction of time overtime hours were used          **
c*      cot    — overtime labor wage per hour                       **
c*      dfsm1  — fraction of time specified material 1 is used      **
c*      dasm1  — average quantity of specified material 1 used per  **
c*               crew day when that material is used                **
c*      csm(1) — unit cost for specified material 1                 **
c*      These three factors, dfsm_, dasm_, and csm are repeated for **
c*              each specified material, 1 through 6.               **
c*                                                                  **
c*    Cost is calculated as follows:                                **
c*                                                                  **
c*    Resource cost per unit of accomplishment =                    **
c*      (# of records * Fraction of time resource is used *         **
c*       Avg quantity of resource when used * Unit cost of resource) **
c*       / Total accomplishment                                     **
c*                                                                  **
c*     The following district cost figures are calculated:          **
c*                                                                  **
c*     dacrh   — average cost for regular labor = (ldacc * dfrh  *  **
c*               adrh * crh) / tdacc                                **
c*     dacot   — average cost for overtime labor = (ldacc * dfot *  **
c*               adot * cot) / tdacc                                **
c*     dacsm1  — average cost for specified material 1 = (ldacc *   **
c*                dfsml * dasml * csm(l)) / tdacc                      **
c*             This material cost calculation is made for each speci- **
c*             fied material, 1 through 6.                            **
c*     NOTE:  In the above calculations, 0.0000000001 is added to     **
c*            denominator to avoid a possible floating divide by zero **
c*            error.                                                  **
c*                                                                    **
c*     daclab  -- average cost for labor = dacrh + dacot              **
c*     dacmat  -- average cost for materials = dacsml + dacsm2 + ...  **
c*                 + dacsm6                                           **
c*     dactot  -- average total cost for materials and labor =        **
c*                daclab + dacmat                                     **
c*                                                                    **
c*   Similar calculations are made for the state as a whole:          **
c*                                                                    **
c*     sacrh   -- average cost for regular labor = (lsacc * sfrh  *   **
c*                asrh * crh) / tsacc                                 **
c*     sacot   -- average cost for overtime labor = (lsacc * sfot *   **
c*                asot * cot) / tsacc                                 **
c*     sacsml  -- average cost for specified material 1 = (lsacc *    **
c*                sfsml * sasml * csm(l)) / tsacc                     **
c*             This material cost calculation is made for each speci- **
c*             fied material, 1 through 6.                            **
c*     NOTE:  In the above calculations, 0.0000000001 is added to     **
c*            denominator to avoid a possible floating divide by zero **
c*            error.                                                  **
c*                                                                    **
c*     saclab  -- average cost for labor = sacrh + sacot              **
c*     sacmat  -- average cost for materials = sacsml + sacsm2 + ...  **
c*                 + sacsm6                                           **
c*     sactot  -- average total cost for materials and labor =        **
c*                saclab + sacmat                                     **
c*                                                                    **
c*          .                                                         **
c**********************************************************************
c**********************************************************************
      dimension csm(6),ldacc(6),tdacc(6),adrh(6),adot(6),
     *      dfrh(6),dfot(6),dfsml(6),dfsm2(6),dfsm3(6),dfsm4(6),
     *    dfsm5(6),dfsm6(6),dasml(6),dasm2(6),dasm3(6),dasm4(6),
     *          dasm5(6),dasm6(6),
     *          dacrh(6),dacot(6),dacsml(6),dacsm2(6),dacsm3(6),
     *          dacsm4(6),dacsm5(6),dacsm6(6),daclab(6),dacmat(6),
     *          dactot(6)
c
c     Cost is calculated as follows:
c     Resource cost per unit of accomplishment =
c       (# of records * Fraction of time resource is used *
c        Avg quantity of resource when used * Unit cost of resource) /
c        Total accomplishment
c
c     Calculate district cost figures
      do 10 j=1,6
      dacrh(j)=(float(ldacc(j)) * dfrh(j) * adrh(j) * crh) / (tdacc(j)
     *         + .0000000001)
      dacot(j)=(float(ldacc(j)) * dfot(j) * adot(j) * cot) / (tdacc(j)
     *         + .0000000001)
      dacsm1(j)=(float(ldacc(j)) * dfsm1(j) * dasm1(j) * csm(1)) /
     *       (tdacc(j)  + .0000000001)
      dacsm2(j)=(float(ldacc(j)) * dfsm2(j) * dasm2(j) * csm(2)) /
     *       (tdacc(j)  + .0000000001)
      dacsm3(j)=(float(ldacc(j)) * dfsm3(j) * dasm3(j) * csm(3)) /
     *       (tdacc(j)  + .0000000001)
      dacsm4(j)=(float(ldacc(j)) * dfsm4(j) * dasm4(j) * csm(4)) /
     *       (tdacc(j)  + .0000000001)
      dacsm5(j)=(float(ldacc(j)) * dfsm5(j) * dasm5(j) * csm(5)) /
     *       (tdacc(j)  + .0000000001)
      dacsm6(j)=(float(ldacc(j)) * dfsm6(j) * dasm6(j) * csm(6)) /
     *       (tdacc(j)  + .0000000001)
      daclab(j)=dacrh(j) + dacot(j)
      dacmat(j)=dacsm1(j) + dacsm2(j) + dacsm3(j) + dacsm4(j) +
     *      dacsm5(j)  + dacsm6(j)
      dactot(j)=daclab(j) + dacmat(j)
 10   continue
c
c     Calculate state cost figures
      sacrh=(float(lsacc) * sfrh * asrh * crh) / (tsacc + .0000000001)
      sacot=(float(lsacc) * sfot * asot * cot) / (tsacc + .0000000001)
      sacsm1=(float(lsacc) * sfsm1 * sasm1 * csm(1)) /
     *       (tsacc + .0000000001)
      sacsm2=(float(lsacc) * sfsm2 * sasm2 * csm(2)) /
     *       (tsacc + .0000000001)
      sacsm3=(float(lsacc) * sfsm3 * sasm3 * csm(3)) /
     *       (tsacc + .0000000001)
      sacsm4=(float(lsacc) * sfsm4 * sasm4 * csm(4)) /
     *       (tsacc + .0000000001)
      sacsm5=(float(lsacc) * sfsm5 * sasm5 * csm(5)) /
     *       (tsacc + .0000000001)
      sacsm6=(float(lsacc) * sfsm6 * sasm6 * csm(6)) /
     *       (tsacc + .0000000001)
      saclab=sacrh + sacot
      sacmat=sacsm1 + sacsm2 + sacsm3 + sacsm4 + sacsm5 + sacsm6
      sactot=saclab + sacmat
      return
      end
      subroutine pdist(iact,dact,uact,ibm,iby,iem,iey,iclass,ldacc,
     *           tdacc,adacc,adcrw,ldrh,dprh,adrh,dfrh,ldot,
     *           dpot,adot,dfot,sm,dfsm1,dfsm2,dfsm3,dfsm4,dfsm5,
     *           dfsm6,dasm1,dasm2,dasm3,dasm4,dasm5,dasm6,dpsm1,
     *           dpsm2,dpsm3,dpsm4,dpsm5,dpsm6,dacrh,dacot,dacsm1,
     *           dacsm2,dacsm3,dacsm4,dacsm5,dacsm6,daclab,dacmat,
     *           dactot)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    PDIST                                                           **
c*                                                                    **
c*    This subroutine prints the district summary information just    **
c*      calculated by DSTLAB, DSTMAT, and DSTCST.  One page contains  **
c*      the labor, material, and cost summary for each of the six     **
c*      IDOH districts.                                               **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      dimension ldacc(6),tdacc(6),adacc(6),adcrw(6),ldrh(6),dprh(6),
     *          adrh(6),dfrh(6),ldot(6),dpot(6),adot(6),dfot(6),
     *          sm(6),dfsm1(6),dfsm2(6),dfsm3(6),dfsm4(6),dfsm5(6),
     *          dfsm6(6),dasm1(6),dasm2(6),dasm3(6),dasm4(6),dasm5(6),
     *          dasm6(6),dpsm1(6),dpsm2(6),dpsm3(6),dpsm4(6),dpsm5(6),
     *          dpsm6(6),dacrh(6),dacot(6),dacsm1(6),dacsm2(6),
     *          dacsm3(6),dacsm4(6),dacsm5(6),dacsm6(6),daclab(6),
     *          dacmat(6),dactot(6)
      character uact*20,dact*40
      integer sm
c
c     Print page header
      write(3,900)iact,dact,ibm,iby,iem,iey,uact
 900  format(1h1,39hROUTINE MAINTENANCE REPORT FOR ACTIVITY,1x,i3,
     *     2x,a40,5x,4hFROM,1x,i2,1h-,i2,1x,7hTHROUGH,1x,i2,1h-,i2,
     *     /,21x,20hACCOMPLISHMENT UNIT:,5x,a20)
c
c     Check to see if printing information for the interstate, other
c       state highway, or total system, and print appropriate heading
      if(iclass.ne.0)go to 10
      write(3,901)
 901  format(/,1x,31hDISTRICT SUMMARY FOR INTERSTATE)
      go to 12
 10   continue
      if(iclass.ne.1)go to 11
      write(3,902)
 902  format(/,1x,40hDISTRICT SUMMARY FOR OTHER STATE HIGHWAY)
      go to 12
 11   continue
      write(3,903)
 903  format(/,1x,45hDISTRICT SUMMARY FOR TOTAL (INT + OSH) SYSTEM)
 12   continue
      write(3,904)
 904  format(////,1x,17hLABOR INFORMATION,/)
```

```
c
c     Print column headings for labor section
      write(3,905)
 905  format(8x,4hCREW,4x,14hACCOMPLISHMENT,5x,3hAVG,6x,
     *        6hTOT RH,18x,6hRH/TOT,4x,6hTOT OT,18x,6hOT/TOT,/,
     *        1x,4hDIST,3x,4hDAYS,4x,5hTOTAL,6x,3hAVG,5x,
     *        4hCREW,6x,4hDAYS,3x,6hRH/ACC,2x,6hAVG RH,3x,4hDAYS,
     *        6x,4hDAYS,3x,6hOT/ACC,2x,6hAVG OT,3x,4hDAYS)
      write(3,906)
 906  format(1x,4h----,2x,6(1h-),2x,9(1h-),2(2x,6(1h-)),4x,6(1h-),
     *        3(2x,6(1h-)),4x,6(1h-),3(2x,6(1h-)))
c
c     Write labor information for each district
      do 13 k=1,6
      write(3,907)k,ldacc(k),tdacc(k),adacc(k),adcrw(k),ldrh(k),
     *              dprh(k),adrh(k),dfrh(k),ldot(k),dpot(k),adot(k),
     *              dfot(k)
 907  format(1x,i4,2x,i6,2x,f9.1,2x,f6.1,2x,f6.2,4x,i6,2(2x,f6.2),
     *        2x,f6.3,4x,i6,2(2x,f6.2),2x,f6.3)
 13   continue
c
c     Print column headings for material section
      write(3,908)
 908  format(////,1x,20hMATERIAL INFORMATION,/)
      write(3,909)(sm(j),j=1,6)
 909  format(6x,4hCREW,1x,6(1x,8h-----MAT,1x,i4,5h-----))
      write(3,910)
 910  format(1x,4hDIST,1x,4hDAYS,1x,6(1x,4hFRAC,1x,6hAVGQNT,1x,
     *        6hQNT/AC))
      write(3,911)
 911  format(1x,4h----,1x,5h-----,6(1x,4h----,2(1x,6(1h-))))
c
c     Write material information for each district
      do 14 k=1,6
      write(3,912)k,ldacc(k),dfsm1(k),dasm1(k),dpsm1(k),dfsm2(k),
     *              dasm2(k),dpsm2(k),dfsm3(k),dasm3(k),dpsm3(k),
     *              dfsm4(k),dasm4(k),dpsm4(k),dfsm5(k),dasm5(k),
     *              dpsm5(k),dfsm6(k),dasm6(k),dpsm6(k)
 912  format(1x,i4,1x,i5,6(1x,f4.2,1x,f6.2,1x,f6.2))
 14   continue
c
c     Write column headings for cost section
      write(3,913)
 913  format(////,1x,16hCOST INFORMATION,5x,
     * 50h(ALL COSTS ARE DOLLARS PER UNIT OF ACCOMPLISHMENT),/)
      write(3,914)
 914  format(8x,4hCREW,5x,5hTOTAL,33x,18(1h-),17hCOSTS BY MATERIAL,
     *        23(1h-))
      write(3,915)(sm(j),j=1,6)
 915  format(1x,4hUNIT,3x,4hDAYS,5x,6hACCOMP,2x,8hTOT COST,2x,
     *        8hLAB COST,2x,8hMAT COST,2x,6(2x,i4,4x))
      write(3,916)
 916  format(1x,4h----,2x,6(1h-),4x,6(1h-),9(2x,8(1h-)))
```

```
c
c     Write cost information for each district
      do 15 j=1,6
      write(3,917)j,ldacc(j),tdacc(j),dactot(j),daclab(j),dacmat(j),
     *            dacsml(j),dacsm2(j),dacsm3(j),dacsm4(j),dacsm5(j),
     *            dacsm6(j)
 917  format(1x,i4,2x,i6,2x,f8.1,9(2x,f8.3))
 15   continue
      return
      end
      subroutine pstate(iact,dact,uact,ibm,iby,iem,iey,iclass,lsacc,
     *           tsacc,asacc,ascrw,lsrh,sprh,asrh,sfrh,lsot,
     *           spot,asot,sfot,sm,sfsml,sfsm2,sfsm3,sfsm4,sfsm5,
     *           sfsm6,sasml,sasm2,sasm3,sasm4,sasm5,sasm6,spsml,
     *           spsm2,spsm3,spsm4,spsm5,spsm6,sacrh,sacot,sacsml,
     *           sacsm2,sacsm3,sacsm4,sacsm5,sacsm6,saclab,sacmat,
     *           sactot)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    PSTATE                                                          **
c*                                                                    **
c*    This subroutine prints the state summary information calculated **
c*      by DSTLAB, MATLAB, and DSTCST.  One page contains the summary **
c*      of labor, materials, and cost for the state as a whole.       **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      dimension sm(6)
      integer sm
      character uact*20,dact*40
c
c     Write page header
      write(3,900)iact,dact,ibm,iby,iem,iey,uact
 900  format(1h1,39hROUTINE MAINTENANCE REPORT FOR ACTIVITY,1x,i3,
     *     2x,a40,5x,4hFROM,1x,i2,1h-,i2,1x,7hTHROUGH,1x,i2,1h-,i2,
     *     /,21x,20hACCOMPLISHMENT UNIT:,5x,a20)
c
c     Check to see if printing information for the interstate, other
c       state highway, or total system, and print appropriate heading
      if(iclass.ne.0)go to 10
      write(3,901)
 901  format(/,1x,28hSTATE SUMMARY FOR INTERSTATE)
      go to 12
 10   continue
      if(iclass.ne.1)go to 11
      write(3,902)
 902  format(/,1x,37hSTATE SUMMARY FOR OTHER STATE HIGHWAY)
      go to 12
 11   continue
      write(3,903)
 903  format(/,1x,42hSTATE SUMMARY FOR TOTAL (INT + OSH) SYSTEM)
 12   continue
```

```
c
c     Print column headings for the labor section
      write(3,904)
 904  format(////,1x,17hLABOR INFORMATION,/)
      write(3,905)
 905  format(8x,4hCREW,4x,14hACCOMPLISHMENT,5x,3hAVG,6x,
     *      6hTOT RH,18x,6hRH/TOT,4x,6hTOT OT,18x,6hOT/TOT,/,
     *      1x,4hDIST,3x,4hDAYS,4x,5hTOTAL,6x,3hAVG,5x,
     *      4hCREW,6x,4hDAYS,3x,6hRH/ACC,2x,6hAVG RH,3x,4hDAYS,
     *      6x,4hDAYS,3x,6hOT/ACC,2x,6hAVG OT,3x,4hDAYS)
      write(3,906)
 906  format(1x,4h----,2x,6(1h-),2x,9(1h-),2(2x,6(1h-)),4x,6(1h-),
     *      3(2x,6(1h-)),4x,6(1h-),3(2x,6(1h-)))
c
c     Print state labor information
      write(3,907)lsacc,tsacc,asacc,ascrw,lsrh,
     *           sprh,asrh,sfrh,lsot,spot,asot,
     *           sfot
 907  format(2x,3hALL,2x,i6,2x,f9.1,2x,f6.1,2x,f6.2,4x,i6,2(2x,f6.2),
     *      2x,f6.3,4x,i6,2(2x,f6.2),2x,f6.3)
c
c     Print headings for the material section
      write(3,908)
 908  format(////,1x,20hMATERIAL INFORMATION,/)
      write(3,909)(sm(j),j=1,6)
 909  format(6x,4hCREW,1x,6(1x,8h-----MAT,1x,i4,5h-----))
      write(3,910)
 910  format(1x,4hDIST,1x,4hDAYS,1x,6(1x,4hFRAC,1x,6hAVGQNT,1x,
     *      6hQNT/AC))
      write(3,911)
 911  format(1x,4h----,1x,5h-----,6(1x,4h----,2(1x,6(1h-))))
c
c     Print material information
      write(3,912)lsacc,sfsm1,sasm1,spsm1,sfsm2,
     *           sasm2,spsm2,sfsm3,sasm3,spsm3,
     *           sfsm4,sasm4,spsm4,sfsm5,sasm5,
     *           spsm5,sfsm6,sasm6,spsm6
 912  format(2x,3hALL,1x,i5,6(1x,f4.2,1x,f6.2,1x,f6.2))
c
c     Print headings for the cost section
      write(3,913)
 913  format(////,1x,16hCOST INFORMATION,5x,
     * 50h(ALL COSTS ARE DOLLARS PER UNIT OF ACCOMPLISHMENT),/)
      write(3,914)
 914  format(8x,4hCREW,5x,5hTOTAL,33x,18(1h-),17hCOSTS BY MATERIAL,
     *      23(1h-))
      write(3,915)(sm(j),j=1,6)
 915  format(1x,4hUNIT,3x,4hDAYS,5x,6hACCOMP,2x,8hTOT COST,2x,
     *      8hLAB COST,2x,8hMAT COST,2x,6(2x,i4,4x))
      write(3,916)
 916  format(1x,4h----,2x,6(1h-),4x,6(1h-),9(2x,8(1h-)))
c
c     Print cost information
```

```
      write(3,917)lsacc,tsacc,sactot,saclab,sacmat,
     *            sacsml,sacsm2,sacsm3,sacsm4,sacsm5,
     *            sacsm6
 917  format(2x,3hALL,2x,i6,2x,f8.1,9(2x,f8.3))
      return
      end
      subroutine devcst(iact,dact,ibm,iby,iem,iey,iclass,
     *                  idev,sdev,cost,nunit,dunit,avg,uact)
c************************************************************************
c************************************************************************
c*                                                                     **
c*    DEVCST                                                           **
c*                                                                     **
c*    This subroutine will identify deviate units based on cost pro-   **
c*      ductivity.  This routine is called only if the user has        **
c*      specified that a deviation analysis is to be conducted.        **
c*      The subroutine takes the average total cost for each subdis-   **
c*      trict, calculates the average and standard deviation, and      **
c*      calculates limits based on the average plus or minus a spec-   **
c*      ified number of standard deviations.  A subdistrict is iden-   **
c*      tified as being deviate if its average cost falls outside      **
c*      these limits.                                                  **
c*                                                                     **
c*    The following variables are used in this subroutine:             **
c*                                                                     **
c*      sdev   - number of standard deviations to be used in deter-    **
c*               mining limits for deviate units                       **
c*      cost   - average cost for each subdistrict                     **
c*      nunit  - number of deviate subdistricts identified             **
c*      dunit  - unit number of a deviate subdistrict                  **
c*      x2     - sum of cost squared                                   **
c*      sx     - sum of cost                                           **
c*      n      - total number of units in calculation of standard de-  **
c*              viation                                                **
c*      var    - variance of average cost                              **
c*      avg    - average of individual subdistrict cost values         **
c*      sd     - standard deviation of average cost                    **
c*      ulimit - upper limit for determining deviate units             **
c*               = avg + (sdev * sd)                                   **
c*      llimit - lower limit for determining deviate units             **
c*               = avg - (sdev * sd)                                   **
c*                                                                     **
c************************************************************************
c************************************************************************
      integer dunit
      character uact*20,dact*40
      real llimit
      dimension cost(66),dunit(20)
c
c     Set summations for cost squared, cost, and number of observations
c       to zero
      x2=0.
      sx=0.
      n=0
c
c     Calculate summation of cost squared, cost, and number of
c       observations over all subdistricts.  Do not include units
c       with a value of zero for cost.
c
c    j keeps track of units numbers
      j1=11
      j2=16
 10   do 11 j=j1,j2
      if(cost(j).eq.0)go to 11
      x2=x2 + cost(j)**2
      sx=sx + cost(j)
      n=n+1
 11   continue
      if(j1.eq.11)go to 20
      if(j1.eq.21)go to 30
      if(j1.eq.31)go to 40
      if(j1.eq.41)go to 50
      if(j1.eq.51)go to 60
      if(j1.eq.61)go to 70
 20   j1=21
      j2=26
      go to 10
 30   j1=31
      j2=36
      go to 10
 40   j1=41
      j2=47
      go to 10
 50   j1=51
      j2=56
      go to 10
 60   j1=61
      j2=66
      go to 10
 70   continue
c
c     To calculate standard deviation, there must be at least 2
c       observations.  If n is less than 2, do not calculate standard
c       deviation.
      if(n.le.1)go to 200
c
c     Calculate variance, standard deviation, average, and limits
      var=(x2 - (sx**2)/n) / (n-1)
      sd=var**.5
      avg=sx/n
      ulimit=avg + sdev * sd
      llimit=avg - sdev * sd
c
c     Find deviate units
      nunit=0
c
```

```
c       j keeps track of unit numbers
        j1=11
        j2=16
  100   do 111 j=j1,j2
        if(cost(j).eq.0)go to 111
        if((cost(j).lt.llimit).or.(cost(j).gt.ulimit))go to 112
        go to 111
  112   nunit=nunit+1
        dunit(nunit)=j
  111   continue
        if(j1.eq.11)go to 120
        if(j1.eq.21)go to 130
        if(j1.eq.31)go to 140
        if(j1.eq.41)go to 150
        if(j1.eq.51)go to 160
        if(j1.eq.61)go to 170
  120   j1=21
        j2=26
        go to 100
  130   j1=31
        j2=36
        go to 100
  140   j1=41
        j2=47
        go to 100
  150   j1=51
        j2=56
        go to 100
  160   j1=61
        j2=66
        go to 100
  170   continue
c
c       Print page listing deviate units
c
c       Print title
        write(3,907)iact,dact,ibm,iby,iem,iey
  907   format(1h1,39hROUTINE MAINTENANCE REPORT FOR ACTIVITY,1x,i3,2x,
       *       a40,5x,4hFROM,1x,i2,1h-,i2,1x,7hTHROUGH,1x,i2,1h-,i2)
c
c       Determine which highway class this is for, and print appropriate
c         heading.
        if(iclass.ne.0)go to 300
        write(3,908)
  908   format(/,1x,29hINTERSTATE DEVIATION ANALYSIS)
        go to 320
  300   continue
        if(iclass.ne.1)go to 310
        write(3,909)
  909   format(/,1x,38hOTHER STATE HIGHWAY DEVIATION ANALYSIS)
        go to 320
  310   continue
        write(3,910)
```

```
 910  format(/,1x,36hTOTAL (INT + OSH) DEVIATION ANALYSIS)
 320  continue
c
c     Print number of standard deviations upon which deviation analysis
c       is based
      write(3,905)sdev
 905  format(/,1x,
     *  52hDEVIATION ANALYSIS BASED ON COST PRODUCTIVITY + OR -,
     *  1x,f6.3,1x,21hSTANDARD DEVIATION(S))
c
c     Print average, standard deviation, and limits
      write(3,901)avg,uact,sd,ulimit,llimit
 901  format(/,1x,21hAVERAGE PRODUCTIVITY=,1x,f7.2,2x,11hDollars per,
     *   1x,a20,/,1x,19hSTANDARD DEVIATION=,3x,f7.2,/,
     *          1x,12hUPPER LIMIT=,10x,f7.2,/,
     *          1x,12hLOWER LIMIT=,10x,f7.2,/)
c
c     Print total number of deviate units
      write(3,902)nunit
 902  format(/,1x,i4,1x,27hDEVIATE UNITS WERE DETECTED)
c
c     Print column headings for list of deviate units
 182  write(3,906)
 906  format(//,1x,4hUNIT,4x,25hPRODUCTIVITY(COST/ACCOMP),/,
     *           1x,4h----,4x,25h-------------------------,/)
 183  continue
c
c     Print unit number and cost value of each deviate subdistrict
      do 200 k=1,nunit
      write(3,904)dunit(k),cost(dunit(k))
 904  format(1x,i2,2h00,4x,f11.2)
 200  continue
      return
      end
      subroutine stdev(val,vmax,avg,sd,am3,am2,aml,apl,ap2,ap3)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*   STDEV                                                            **
c*                                                                    **
c*   This subroutine will calculate the average, standard deviation,  **
c*     average  plus and minus 1, 2, and 3 standard deviations, and   **
c*    the maximum value of a group of numbers.  This information will **
c*    be used to make a bar chart.                                    **
c*                                                                    **
c*   The following variables are used:                                **
c*                                                                    **
c*     val  - values being averaged                                   **
c*     x2   - sum of val squared values                               **
c*     sx   - sum of val values                                       **
c*     n    - number of observations (number of val values)           **
c*     vmax - maximum value                                           **
c*     avg  - average value of non-zero val's                         **
c*     sd   - standard deviation                                      **
c*     am3  - avg - 3 * sd                                            **
c*     am2  - avg - 2 * sd                                            **
c*     aml  - avg - 1 * sd                                            **
c*     apl  - avg + 1 * sd                                            **
c*     ap2  - avg + 2 * sd                                            **
c*     ap3  - avg + 3 * sd                                            **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      dimension val(66)
c
c     Initialize s2, sx, n, and vmax to zero
      x2=0.
      sx=0.
      n=0.
      vmax=0.
c
c     Find vmax, and calculate s2, sx, and n, using nonzero val values
c
c     j keeps track of subdistricts
      jl=11
      j2=16
 10   do 12 j=jl,j2
      if(vmax.ge.val(j))go to 11
      vmax=val(j)
 11   continue
      if(val(j).eq.0.)go to 12
      x2=x2+val(j)**2
      sx=sx+val(j)
      n=n+1
 12   continue
      if(jl.eq.11)go to 20
      if(jl.eq.21)go to 30
      if(j1.eq.31)go to 40
      if(j1.eq.41)go to 50
      if(j1.eq.51)go to 60
      if(j1.eq.61)go to 70
 20   j1=21
      j2=26
      go to 10
 30   j1=31
      j2=36
      go to 10
 40   j1=41
      j2=47
      go to 10
 50   j1=51
      j2=56
      go to 10
 60   j1=61
      j2=66
      go to 10
 70   continue
c
c     If n is less than two, standard deviation cannot be calculated.
c       Check for this condition.
      if(n.le.1)go to 100
c
c     Calculate variance, standard deviation, average, and average
c       plus or minus 1, 2, and 3 standard deviations
      var=(x2-(sx**2)/n)/(n-1)
      sd=var ** .5
      avg=sx/n
      am3=avg-3*sd
      am2=avg-2*sd
      am1=avg-sd
      ap1=avg+sd
      ap2=avg+2*sd
      ap3=avg+3*sd
 100  continue
      return
      end
      subroutine bar(iact,dact,uact,ibm,iby,iem,iey,iclass,ichart,
     *               sm,usm,actot,tacc,aacc,val,vmax,
     *               avg,sd,am3,am2,am1,ap1,ap2,ap3)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    BAR                                                             **
c*                                                                    **
c*    This subroutine will print a bar chart of average cost per      **
c*      production unit for all subdistricts.  In addition to the bar **
c*      chart of average cost, the values for averge cost, total num- **
c*      ber of crew days, and average and total accomplishment are    **
c*      printed.  On the bar chart, the average value, and the aver-  **
c*      plus and minus a number of standard deviations are indicated. **
```

```
c*                                                                    **
c*   The bar chart is 75 characters long, and is scaled such that    **
c*     the bar for the unit (subdistrict) with the highest value is  **
c*     75 characters long.  On the first and last lines of the       **
c*     chart, the locations of the average and the average plus or   **
c*     minus up to three standard deviations are indicated.  The     **
c*     width of the range in terms of standard deviations depends on **
c*     the particular values involved.  If any of the limits (A lim- **
c*     it used here refers to a value of the average plus or minus a **
c*     number of standard deviations.) are less than zero or greater **
c*     than the largest value for a subdistrict, those limits are    **
c*     not plotted on the chart.  A number of tests are required to  **
c*     determine which set of limits may be included in the chart,   **
c*     and call the appropriate format statement.  Also, since all   **
c*     values must be rounded to the nearest integer for plotting    **
c*     (A fraction of a character cannot be printed.) the possibili- **
c*     ty of round-off errors must be considered.                    **
c*                                                                    **
c*   Up to six limits may be printed along with the average:         **
c*                                                                    **
c*    Lower limits                                                    **
c*     am3 - Average minus three standard deviations                  **
c*     am2 - Average minus two standard deviations                    **
c*     aml - Average minus one standard deviations                    **
c*    Upper limits                                                    **
c*     apl - Average plus one standard deviations                     **
c*     ap2 - Average plus two standard deviations                     **
c*     ap3 - Average plus three standard deviations                   **
c*                                                                    **
c*     First, the smallest positive lower limit is determined.        **
c*                                                                    **
c*   |         3    2    1    A    1    2    3          |             **
c*                                                                    **
c*   |<-- bm -->|<--------- 6b + 7 ---------->|<-- bl -->|            **
c*                                                                    **
c*   bm = number of spaces from left end of chart (zero) to first    **
c*        limit, in this case, am3                                    **
c*   b  = number of spaces less one required for one standard devia- **
c*        tion (nint((sd/vmax)*75.)-1)                                **
c*   bl = number of spaces between last limit and right end of chart **
c*   A  = location of average                                         **
c*   3  = three standard deviations from average                      **
c*   2  = two standard deviations from average                        **
c*   1  = one standard deviation from average                         **
c*                                                                    **
c*   First, the program calculates the value of b.  This is the num- **
c*     ber of spaces needed to represent one standard deviation,     **
c*     less one space which is used by the character 1, 2, or 3.     **
c*     Next the program determines which is the smallest positive    **
c*     lower limit, and bm is calculated.  Because of the rounding   **
c*     done, bm could be negative.  If this is the case, the limits  **
c*     will not be plotted correctly, so the next highest lower lim- **
c*     it will be used and bm calculated for this limit.             **
```

```
c*      Next, the highest upper limit that is within the chart range    **
c*      is determined.  The chart range is from 0 to vmax, where vmax   **
c*      is the highest individual unit value.  Once this is determin-   **
c*      ed, bl is calculated.  Again, due to rounding, bl could have    **
c*      a negative value.  If this is the case, make an adjustment by   **
c*      decreasing bm by bl and setting bl equal to zero.  If this      **
c*      adjustment is very large, it could cause bm to be negative.     **
c*      If this happens, move on two use the next highest lower limit.**
c*                                                                      **
c*    When all the correct scaling has been made, the line showing      **
c*      the limits is plotted according to the particular format        **
c*      required.  This format is indicated by the value of wcode.      **
c*      Wcode is used to repeat the limit line after the bar for each   **
c*      unit has been plotted.                                          **
c*                                                                      **
c*                                                                      **
c*************************************************************************
c*************************************************************************
      integer wcode,b,bm,bl,sm
      character x,blank,dact*40,uact*20,usm*20,s1,s2,s3,sa
      dimension actot(66),x(75),blank(75),val(66),tacc(66),aacc(66)
      lwrite=0
      s1='1'
      s2='2'
      s3='3'
      sa='A'
      do 10 k=1,75
      x(k)='X'
      blank(k)=' '
 10   continue
c
c     Write heading for chart
      write(3,900)iact,dact,ibm,iby,iem,iey
 900  format(1h1,39hROUTINE MAINTENANCE REPORT FOR ACTIVITY,1x,i3,2x,
     *       a40,5x,4hFROM,1x,i2,1h-,i2,1x,7hTHROUGH,1x,i2,1h-,i2)
c
c     Determine which highway class the chart is for and print
c       appropriate heading
      if(iclass.ne.0)go to 11
      write(3,901)
 901  format(/,1x,10hINTERSTATE,/)
      go to 13
 11   continue
      if(iclass.ne.1)go to 12
      write(3,902)
 902  format(/,1x,19hOTHER STATE HIGHWAY,/)
      go to 13
 12   continue
      write(3,903)
 903  format(/,1x,24hTOTAL (INT + OSH) SYSTEM,/)
 13   continue
c
c     Write the average and standard deviation values at the top of
```

```
c       the chart
c     Because there are five different charts, the subroutine to print
c       the top portion of the particular chart desired is called.
      if(ichart.ne.0)go to 300
      call csttop(avg,sd,uact)
 300  continue
      if(ichart.ne.1)go to 301
      call hrstop(avg,sd,uact)
 301  continue
      if(ichart.ne.2)go to 302
      call tactop(avg,sd,uact)
 302  continue
      if(ichart.ne.3)go to 303
      call aactop(avg,sd,uact)
 303  continue
      if(ichart.ne.4)go to 304
      call crwtop(avg,sd,uact)
 304  continue
      if(ichart.ne.5)go to 305
      call mattop(avg,sd,uact,sm,usm)
 305  continue
c
c     Print line indicating position of avg, and limits +/- 1,2,3 sd.
c       Before printing, need to know if any limits are outside the
c       range of 0 to vmax.
c     Check lower limits (am3,am2,am1).  for each lower limit
c      condition, there are four upper limit  (ap1,ap2,ap3) conditions.
      b=nint((sd/vmax)*75.)-1
      if(am3.lt.0)go to 30
c
c     In this case, am3 is positive.  Check ap1,ap2,ap3.
      bm=nint(((avg-3.*sd)/vmax)*75.)-1
c
c     If bm is negative, the correct limit lines will not be
c       printed because of the rounding adjustments made.
c       If bm is negative, go to the next highest lower limit.
c       This check is needed until the lower limit is the average.
      if(bm.lt.0)go to 30
 20   if(ap3.gt.vmax)go to 21
c
c     In this case, ap3 is less than vmax which is the right end
c       of the chart
c
c     wcode=1 indicates that the limits am3 thorugh ap3 are on the chart
      wcode=1
      bl=75-bm-7-6*b
c
c     Because of possible round off errors, bl could be negative.
c       Check for this.  If bl is negative, make an adjustment in
c       bm.
      if(bl.ge.0)go to 24
      bm=bm + bl
      if(bm.lt.0)go to 30
      bl=0
 24   continue
c
c     Print line indicating limits
      write(3,910)(blank(i),i=1,bm),s3,(blank(i),i=1,b),s2,
     *            (blank(i),i=1,b),s1,(blank(i),i=1,b),sa,
     *            (blank(i),i=1,b),s1,(blank(i),i=1,b),s2,
     *            (blank(i),i=1,b),s3,(blank(i),i=1,bl)
 910  format(1x,1hI,8x,75a1,1x,1hI,35x,1hI)
      go to 100
 21   continue
c
c     Check if ap2 is within chart limits
      if(ap2.gt.vmax)go to 22
c
c     In this case, ap3 is not within chart limit (vmax), but ap2 is.
c
c     wcode=2 indicates that am3 through ap2 are within chart limits
      wcode=2
      bl=75-bm-6-5*b
      if(bl.ge.0)go to 25
      bm=bm + bl
      if(bm.lt.0)go to 30
      bl=0
 25   continue
      write(3,910)(blank(i),i=1,bm),s3,(blank(i),i=1,b),s2,
     *            (blank(i),i=1,b),s1,(blank(i),i=1,b),sa,
     *            (blank(i),i=1,b),s1,(blank(i),i=1,b),s2,
     *            (blank(i),i=1,bl)
      go to 100
 22   continue
      if(ap1.gt.vmax)go to 23
c
c     In this case, ap1 is the largest limit within chart limit (vmax).
c
c     wcode=3 indicates that am3 thorough ap1 are within chart limits
      wcode=3
      bl=75-bm-5-4*b
      if(bl.ge.0)go to 26
      bm=bm + bl
      if(bm.lt.0)go to 30
      bl=0
 26   continue
      write(3,910)(blank(i),i=1,bm),s3,(blank(i),i=1,b),s2,
     *            (blank(i),i=1,b),s1,(blank(i),i=1,b),sa,
     *            (blank(i),i=1,b),s1,(blank(i),i=1,bl)
      go to 100
 23   continue
c
c     In this case, ap1 is beyond vmax
c
c     wcode=4 indicates that am3 through average are within chart limits
      wcode=4
      bl=75-bm-4-3*b
      if(bl.ge.0)go to 27
      bm=bm + bl
      if(bm.lt.0)go to 30
      bl=0
 27   continue
      write(3,910)(blank(i),i=1,bm),s3,(blank(i),i=1,b),s2,
     *             (blank(i),i=1,b),s1,(blank(i),i=1,b),sa,
     *             (blank(i),i=1,bl)
      go to 100
 30   continue
c
c     In this case, am3 is less than zero, check am2.
      if(am2.lt.0.)go to 40
c
c     In this case, the lower limit is am2; check ap1,ap2,ap3.
      bm=nint(((avg-2.*sd)/vmax)*75.)-1
      if(bm.lt.0)go to 40
 34   if(ap3.gt.vmax)go to 31
c
c     Limits within chart are am2 through ap3
      wcode=5
      bl=75-bm-6-5*b
      if(bl.ge.0)go to 35
      bm=bm + bl
      if(bm.lt.0)go to 40
      bl=0
 35   continue
      write(3,910)(blank(i),i=1,bm),s2,(blank(i),i=1,b),s1,
     *             (blank(i),i=1,b),sa,(blank(i),i=1,b),s1,
     *             (blank(i),i=1,b),s2,(blank(i),i=1,b),s3,
     *             (blank(i),i=1,bl)
      go to 100
 31   if(ap2.gt.vmax)go to 32
c
c     Limits within chart are am2 through ap2
      wcode=6
      bl=75-bm-5-4*b
      if(bl.ge.0)go to 36
      bm=bm+bl
      if(bm.lt.0)go to 40
      bl=0
 36   continue
      write(3,910)(blank(i),i=1,bm),s2,(blank(i),i=1,b),s1,
     *             (blank(i),i=1,b),sa,(blank(i),i=1,b),s1,
     *             (blank(i),i=1,b),s2,(blank(i),i=1,bl)
      go to 100
 32   if(ap1.gt.vmax)go to 33
c
c     Limits within chart are am2 through ap1
      wcode=7
      bl=75-bm-4-3*b
      if(bl.ge.0)go to 37
      bm=bm + bl
      if(bm.lt.0)go to 40
      bl=0
 37   continue
      write(3,910)(blank(i),i=1,bm),s2,(blank(i),i=1,b),s1,
     *             (blank(i),i=1,b),sa,(blank(i),i=1,b),s1,
     *             (blank(i),i=1,bl)
      go to 100
c
c     Limits within chart are am2 through average
 33   wcode=8
      bl=75-bm-3-2*b
      if(bl.ge.0)go to 38
      bm=bm + bl
      if(bm.lt.0)go to 40
      bl=0
 38   continue
      write(3,910)(blank(i),i=1,bm),s2,(blank(i),i=1,b),s1,
     *             (blank(i),i=1,b),sa,(blank(i),i=1,bl)
      go to 100
 40   continue
      if(am1.lt.0.)go to 50
      bm=nint(((avg-sd)/vmax)*75.)-1
      if(bm.lt.0)go to 50
 44   if(ap3.gt.vmax)go to 41
c
c     Limits within chart are am1 through ap3
      wcode=9
      bl=75-bm-5-4*b
      if(bl.ge.0)go to 45
      bm=bm+bl
      if(bm.lt.0)go to 50
      bl=0
 45   continue
      write(3,910)(blank(i),i=1,bm),s1,(blank(i),i=1,b),sa,
     *             (blank(i),i=1,b),s1,(blank(i),i=1,b),s2,
     *             (blank(i),i=1,b),s3,(blank(i),i=1,bl)
      go to 100
 41   continue
      if(ap2.gt.vmax)go to 42
c
c     Limits within chart are am1 through ap2
      wcode=10
      bl=75-bm-4-3*b
      if(bl.ge.0)go to 46
      bm=bm+bl
      if(bm.lt.0)go to 50
      bl=0
 46   continue
      write(3,910)(blank(i),i=1,bm),s1,(blank(i),i=1,b),sa,
     *             (blank(i),i=1,b),s1,(blank(i),i=1,b),s2,
     *             (blank(i),i=1,bl)
      go to 100
 42   continue
      if(ap1.gt.vmax)go to 43
c
c     Limits within chart are am1 through ap1
      wcode=11
      b1=75-bm-3-2*b
      if(b1.ge.0)go to 47
      bm=bm+b1
      if(bm.1t.0)go to 50
      b1=0
 47   continue
      write(3,910)(blank(i),i=1,bm),s1,(blank(i),i=1,b),sa,
     *              (blank(i),i=1,b),s1,(blank(i),i=1,b1)
      go to 100
 43   continue
c
c     Limits within chart are am1 through average
      wcode=12
      b1=75-bm-2-b
      if(b1.ge.0)go to 48
      bm=bm+b1
      if(bm.1t.0)go to 50
      b1=0
 48   continue
      write(3,910)(blank(i),i=1,bm),s1,(blank(i),i=1,b),sa,
     *              (blank(i),i=1,b1)
      go to 100
 50   continue
      bm=nint((avg/vmax)*75.)-1
 54   if(ap3.gt.vmax)go to 51
c
c     Limits within chart are average through ap3
      wcode=13
      b1=75-bm-4-3*b
      if(b1.ge.0)go to 55
      bm=bm+b1
      b1=0
 55   continue
      write(3,910)(blank(i),i=1,bm),sa,(blank(i),i=1,b),s1,
     *              (blank(i),i=1,b),s2,(blank(i),i=1,b),s3,
     *              (blank(i),i=1,b1)
      go to 100
 51   continue
      if(ap2.gt.vmax)go to 52
c
c     Limits within chart are average through ap2
      wcode=14
      b1=75-bm-3-2*b
      if(b1.ge.0)go to 56
      bm=bm+b1
      b1=0
 56   continue
      write(3,910)(blank(i),i=1,bm),sa,(blank(i),i=1,b),s1,
     *              (blank(i),i=1,b),s2,(blank(i),i=1,bl)
      go to 100
 52   continue
      if(apl.gt.vmax)go to 53
c
c     Limits within chart are average through apl
      wcode=15
      bl=75-bm-2-b
      if(bl.ge.0)go to 57
      bm=bm+bl
      bl=0
 57   continue
      write(3,910)(blank(i),i=1,bm),sa,(blank(i),i=1,b),sl,
     *              (blank(i),i=1,bl)
      go to 100
 53   continue
c
c     Only limit within chart limits is average
      wcode=16
      bl=75-bm-1
      if(bl.ge.0)go to 58
      bm=bm+bl
      bl=0
 58   continue
      write(3,910)(blank(i),i=1,bm),sa,(blank(i),i=1,bl)
 100  continue
      if(lwrite.eq.1)go to 200
c
c     Print bar for each unit.  Because the format for each of the
c       charts is slightly different, a particular subroutine will
c       print the bars and data for each chart.  Note that labor
c       hours and crew size use the same format and subroutine.
      if(ichart.ne.0)go to 110
      call barcst(vmax,x,blank,actot,val,tacc,aacc)
 110  continue
      if(ichart.ne.1)go to 120
      call barhrs(vmax,x,blank,val,actot,tacc,aacc)
 120  continue
      if(ichart.ne.2)go to 130
      call bartac(vmax,x,blank,tacc,aacc,val,actot)
 130  continue
      if(ichart.ne.3)go to 140
      call baraac(vmax,x,blank,aacc,tacc,val,actot)
 140  continue
      if(ichart.ne.4)go to 150
      call barhrs(vmax,x,blank,val,actot,tacc,aacc)
 150  continue
      if(ichart.ne.5)go to 160
      call barhrs(vmax,x,blank,val,actot,tacc,aacc)
 160  continue
      lwrite=1
c
c     Print limits along bottom.
```

```
      if(wcode.eq.1)go to 20
      if(wcode.eq.2)go to 21
      if(wcode.eq.3)go to 22
      if(wcode.eq.4)go to 23
      if(wcode.eq.5)go to 34
      if(wcode.eq.6)go to 31
      if(wcode.eq.7)go to 32
      if(wcode.eq.8)go to 33
      if(wcode.eq.9)go to 44
      if(wcode.eq.10)go to 41
      if(wcode.eq.11)go to 42
      if(wcode.eq.12)go to 43
      if(wcode.eq.13)go to 54
      if(wcode.eq.14)go to 51
      if(wcode.eq.15)go to 52
      if(wcode.eq.16)go to 53
 200  continue
c
c     Print bottom border of chart
      write(3,912)
 912  format(1x,122(1h-))
c
c     Print explanation of symbols
      write(3,913)
 913  format(/,10x,12hA — Average,/,
     *       10x,40h1 — One Standard Deviation From Average,/,
     *      10x,41h2 — Two Standard Deviations From Average,/,
     *    10x,43h3 — Three Standard Deviations From Average)
      return
      end
      subroutine csttop(avg,sd,uact)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    CSTTOP                                                          **
c*                                                                    **
c*    This subroutine prints the top lines of the bar chart of aver-  **
c*      age cost.                                                     **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      character uact*20
c
c     Write the average and standard deviation values at the top of
c       the chart
      write(3,900)avg,uact,sd,uact
 900  format(1x,13hAverage Cost=,7x,f7.2,2x,11hDollars per,1x,a20,/,1x,
     *       19hStandard Deviation=,1x,f7.2,2x,11hDollars per,1x,a20,//)
c
c     Print top border of chart
      write(3,901)
 901  format(1x,122(1h-))
c
```

```
c     Print chart headings
      write(3,902)
 902  format(1x,1hI,84x,1hI,1x,7hAVERAGE,3x,4hCREW,4x,
     *        14hACCOMPLISHMENT,2x,1hI)
      write(3,903)uact
 903  format(1x,1hI,1x,4hUNIT,10x,
     *        39hChart Shows Average Cost in Dollars per,
     * 1x,a20,9x,1hI,2x,4hCOST,5x,4hDAYS,4x,5hTOTAL,3x,7hAVERAGE,
     * 1x,1hI,/,1x,1hI,1x,4h----,3x,76(1h-),1hI,1x,7(1h-),2x,6(1h-),
     * 2x,8(1h-),1x,7(1h-),1x,1hI)
      return
      end
      subroutine hrstop(avg,sd,uact)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    HRSTOP                                                          **
c*                                                                    **
c*    This subroutine prints the top part of the bar chart of average **
c*      labor hours per accomplishment unit.                          **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      character uact*20
      write(3,900)avg,uact,sd,uact
 900  format(1x,37hAverage Labor Hours (Reg + Overtime)=,2x,f6.2,2x,
     *        9hHours per,1x,a20,/,1x,
     *        19hStandard Deviation=,20x,f6.2,2x,9hHours per,1x,a20,//)
      write(3,901)
 901  format(1x,122(1h-))
      write(3,902)
 902  format(1x,1hI,84x,1hI,1x,6hHOURS/,2x,7hAVERAGE,3x,
     *        14hACCOMPLISHMENT,2x,1hI)
      write(3,903)uact
 903  format(1x,1hI,1x,4hUNIT,5x,
     *        52hChart Shows Average Labor Hours (Reg + Overtime) per,
     *        1x,a20,1x,1hI,1x,6hACCOMP,3x,4hCOST,5x,5hTOTAL,3x,
     *        7hAVERAGE,1x,1hI,/,1x,1hI,1x,4h----,3x,76(1h-),1hI,
     *        1x,6(1h-),2x,7(1h-),2x,8(1h-),1x,7(1h-),1x,1hI)
      return
      end
      subroutine tactop(avg,sd,uact)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    TACTOP                                                          **
c*                                                                    **
c*    This subroutine prints the top lines of the bar chart of total  **
c*      accomplishment.                                               **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      character uact*20
      write(3,900)avg,uact,sd,uact
 900  format(1x,29hAverage Total Accomplishment=,1x,f8.1,2x,a20,/,1x,
     *       19hStandard Deviation=,13x,f6.1,2x,a20,//)
      write(3,901)
 901  format(1x,122(1h-))
      write(3,902)
 902  format(1x,1hI,84x,1hI,2x,14hACCOMPLISHMENT,4x,4hCREW,3x,
     *       7hAVERAGE,1x,1hI)
      write(3,903)uact
 903  format(1x,1hI,1x,4hUNIT,12x,
     *  35hChart Shows Total Accomplishment in,1x,a20,11x,1hI,2x,
     *  5hTOTAL,3x,7hAVERAGE,3x,4hDAYS,4x,4hCOST,3x,1hI,/,
     *  1x,1hI,1x,4h----,3x,76(1h-),1hI,1x,8(1h-),1x,7(1h-),2x,
     *  6(1h-),2x,7(1h-),1x,1hI)
      return
      end
      subroutine aactop(avg,sd,uact)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*   AACTOP                                                           **
c*                                                                    **
c*   This subroutine prints the top part of the bar chart of average  **
c*     daily accomplishment.                                          **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      character uact*20
      write(3,900)avg,uact,sd,uact
 900  format(1x,29hAverage Daily Accomplishment=,1x,f8.1,2x,a20,/,1x,
     *       19hStandard Deviation=,13x,f6.1,2x,a20,//)
      write(3,901)
 901  format(1x,122(1h-))
      write(3,902)
 902  format(1x,1hI,84x,1hI,2x,14hACCOMPLISHMENT,4x,4hCREW,3x,
     *       7hAVERAGE,1x,1hI)
      write(3,903)uact
 903  format(1x,1hI,1x,4hUNIT,8x,
     * 43hChart Shows Average Daily Accomplishment in,1x,a20,7x,1hI,
     * 1x,7hAVERAGE,3x,5hTOTAL,4x,4hDAYS,4x,4hCOST,3x,1hI,/,1x,1hI,1x,
     * 4h----,3x,76(1h-),1hI,1x,7(1h-),1x,8(1h-),2x,6(1h-),2x,7(1h-),
     * 1x,1hI)
      return
      end
      subroutine crwtop(avg,sd,uact)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    CRWTOP                                                          **
c*                                                                    **
c*    This subroutine prints the top part of the bar chart of average **
c*      crew size.                                                    **
c*                                                                    **
```

```
c************************************************************************
c************************************************************************
      character uact*20
      write(3,900)avg,sd
 900  format(1x,18hAverage Crew Size=,2x,f7.2,2x,7hPersons,/,1x,
     *       19hStandard Deviation=,1x,f7.2,2x,7hPersons,//)
      write(3,901)
 901  format(1x,122(1h-))
      write(3,902)
 902  format(1x,1hI,84x,1hI,2x,4hCREW,3x,7hAVERAGE,3x,
     *       14hACCOMPLISHMENT,2x,1hI)
      write(3,903)
 903  format(1x,1hI,1x,4hUNIT,9x,
     * 40hChart Shows Average Crew Size in Persons,21x,9x,1hI,2x,
     * 4hSIZE,4x,4hCOST,5x,5hTOTAL,3x,7hAVERAGE,1x,1hI,/,1x,1hI,1x,
     * 4h----,3x,76(1h-),1hI,1x,6(1h-),2x,7(1h-),2x,8(1h-),1x,7(1h-),
     * 1x,1hI)
      return
      end
      subroutine mattop(avg,sd,uact,sm,usm)
c************************************************************************
c************************************************************************
c*                                                                     **
c*    MATTOP                                                           **
c*                                                                     **
c*    This subroutine prints the top part of the bar chart of average **
c*      material quantity per accomplishment unit.                     **
c*                                                                     **
c************************************************************************
c************************************************************************
      character uact*20,usm*20
      integer sm
      write(3,900)sm,avg,usm,uact,sd,usm,uact
 900  format(1x,28hAverage Quantity of Material,1x,i4,1h=,f7.2,1x,a20,
     *       1x,3hper,1x,a20,/,1x,19hStandard Deviation=,15x,f7.2,1x,
     *       a20,1x,3hper,1x,a20,//)
      write(3,901)
 901  format(1x,122(1h-))
      write(3,902)
 902  format(1x,1hI,84x,1hI,1x,5hQUAN/,
     *       3x,7hAVERAGE,3x,14hACCOMPLISHMENT,2x,1hI)
      write(3,903)sm
 903  format(1x,1hI,1x,4hUNIT,3x,
     *      40hChart Shows Average Quantity of Material,1x,
     *      i4,1x,3hper,1x,19hAccomplishment Unit,7x,1hI,1x,6hACCOMP,
     *      3x,4hCOST,5x,5hTOTAL,3x,7hAVERAGE,1x,1hI,/,1x,1hI,1x,
     * 4h----,3x,76(1h-),1hI,1x,6(1h-),2x,7(1h-),2x,8(1h-),1x,7(1h-),
     * 1x,1hI)
      return
      end
      subroutine barcst(vmax,x,blank,actot,lacc,tacc,aacc)
c************************************************************************
c************************************************************************
```

```
c*                                                                    **
c*    BARCST                                                          **
c*                                                                    **
c*    This subroutine will print the bar and values of average cost,  **
c*      number of crew days, total and average accomplishment.        **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      dimension x(75),blank(75),actot(66),lacc(66),tacc(66),aacc(66)
      character x,blank
c
c     Print bar for each unit
      j1=11
      j2=16
 110  do 111 j=j1,j2
      ibar=nint((actot(j)/vmax)*75)
      iblank=75-ibar
      write(3,911)j,(x(i),i=1,ibar),(blank(i),i=1,iblank),actot(j),
     *           lacc(j),tacc(j),aacc(j)
 911  format(1x,1hI,1x,i2,2h00,3x,75a1,1x,1hI,1x,f7.2,2x,i6,2x,
     *        f8.1,2x,f6.1,1x,1hI)
 111  continue
      if(j1.eq.11)go to 120
      if(j1.eq.21)go to 130
      if(j1.eq.31)go to 140
      if(j1.eq.41)go to 150
      if(j1.eq.51)go to 160
      if(j1.eq.61)go to 170
 120  j1=21
      j2=26
      go to 110
 130  j1=31
      j2=36
      go to 110
 140  j1=41
      j2=47
      go to 110
 150  j1=51
      j2=56
      go to 110
 160  j1=61
      j2=66
      go to 110
 170  continue
      return
      end
      subroutine barhrs(vmax,x,blank,val,actot,tacc,aacc)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    BARHRS                                                          **
c*                                                                    **
c*    This subroutine will print the bar and values of labor hours,   **
```

```
c*      average cost, total and average accomplishment.                 **
c*      It also prints the bars for the average crew size chart.        **
c*                                                                      **
c***********************************************************************
c***********************************************************************
      dimension x(75),blank(75),actot(66),val(66),tacc(66),aacc(66)
      character x,blank
c     Print bar for each unit
      j1=11
      j2=16
 110  do 111 j=j1,j2
      ibar=nint((val(j)/vmax)*75)
      iblank=75-ibar
      write(3,911)j,(x(i),i=1,ibar),(blank(i),i=1,iblank),val(j),
     *            actot(j),tacc(j),aacc(j)
 911  format(1x,1hI,1x,i2,2h00,3x,75a1,1x,1hI,1x,f6.2,2x,f7.2,2x,
     *       f8.1,2x,f6.1,1x,1hI)
 111  continue
      if(j1.eq.11)go to 120
      if(j1.eq.21)go to 130
      if(j1.eq.31)go to 140
      if(j1.eq.41)go to 150
      if(j1.eq.51)go to 160
      if(j1.eq.61)go to 170
 120  j1=21
      j2=26
      go to 110
 130  j1=31
      j2=36
      go to 110
 140  j1=41
      j2=47
      go to 110
 150  j1=51
      j2=56
      go to 110
 160  j1=61
      j2=66
      go to 110
 170  continue
      return
      end
      subroutine bartac(vmax,x,blank,tacc,aacc,lacc,actot)
c***********************************************************************
c***********************************************************************
c*                                                                      **
c*    BARTAC                                                            **
c*                                                                      **
c*    This subroutine will print the bar and values of total accomp-   **
c*      lishment, average accomplishment, number of crew days, and     **
c*      average cost.                                                   **
c*                                                                      **
c***********************************************************************
```

```
c***********************************************************************
      dimension x(75),blank(75),actot(66),lacc(66),tacc(66),aacc(66)
      character x,blank
c     Print bar for each unit
      j1=11
      j2=16
 110  do 111 j=j1,j2
      ibar=nint((tacc(j)/vmax)*75)
      iblank=75-ibar
      write(3,911)j,(x(i),i=1,ibar),(blank(i),i=1,iblank),tacc(j),
     *            aacc(j),lacc(j),actot(j)
 911  format(1x,1hI,1x,i2,2h00,3x,75a1,1x,1hI,1x,f8.1,2x,f6.1,2x,
     *       i6,2x,f7.2,1x,1hI)
 111  continue
      if(j1.eq.11)go to 120
      if(j1.eq.21)go to 130
      if(j1.eq.31)go to 140
      if(j1.eq.41)go to 150
      if(j1.eq.51)go to 160
      if(j1.eq.61)go to 170
 120  j1=21
      j2=26
      go to 110
 130  j1=31
      j2=36
      go to 110
 140  j1=41
      j2=47
      go to 110
 150  j1=51
      j2=56
      go to 110
 160  j1=61
      j2=66
      go to 110
 170  continue
      return
      end
      subroutine baraac(vmax,x,blank,aacc,tacc,lacc,actot)
c***********************************************************************
c***********************************************************************
c*                                                                    **
c*    BARAAC                                                          **
c*                                                                    **
c*    This subroutine will print the bar for average accomplishment, **
c*      and values of average and total accomplishment, number of    **
c*      crew days, and average cost.                                  **
c*                                                                    **
c***********************************************************************
c***********************************************************************
      dimension x(75),blank(75),actot(66),lacc(66),tacc(66),aacc(66)
      character x,blank
c
```

```
c     Print bar for each unit
      j1=11
      j2=16
 110  do 111  j=j1,j2
      ibar=nint((aacc(j)/vmax)*75)
      iblank=75-ibar
      write(3,911)j,(x(i),i=1,ibar),(blank(i),i=1,iblank),aacc(j),
     *            tacc(j),lacc(j),actot(j)
 911  format(1x,1hI,1x,i2,2h00,3x,75a1,1x,1hI,2x,f6.1,1x,f8.1,2x,
     *       i6,2x,f7.2,1x,1hI)
 111  continue
      if(j1.eq.11)go to 120
      if(j1.eq.21)go to 130
      if(j1.eq.31)go to 140
      if(j1.eq.41)go to 150
      if(j1.eq.51)go to 160
      if(j1.eq.61)go to 170
 120  j1=21
      j2=26
      go to 110
 130  j1=31
      j2=36
      go to 110
 140  j1=41
      j2=47
      go to 110
 150  j1=51
      j2=56
      go to 110
 160  j1=61
      j2=66
      go to 110
 170  continue
      return
      end
```

COVER DESIGN BY ALDO GIORGINI